Automatic road network extraction in suburban areas from aerial images

Von der Fakultät für Bauingenieurwesen und Geodäsie
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades

DOKTOR-INGENIEUR (Dr.-Ing.)

genehmigte Dissertation
von

Dipl.-Ing. Anne Grote

geboren am 22.09.1979 in Berlin

2011

Vorsitzender der Prüfungskommission: Prof. Dr.-Ing. Hansjörg Kutterer
Referenten: Prof. Dr.-Ing. Christian Heipke
Prof. Dr.-Ing. Markus Gerke
PD Dr.techn. Franz Rottensteiner
Prof. Dr.-Ing. Monika Sester

Tag der mündlichen Prüfung: 08. April 2011

Die Dissertation wurde am 04. Januar 2011 bei der
Fakultät für Bauingenieurwesen und Geodäsie der Gottfried Wilhelm Leibniz Universität Hannover
eingereicht.

Summary

In this thesis, a new method for the extraction of road networks in suburban areas from optical aerial images is developed. The road extraction method is region-based; road regions are extracted from a segmented image and combined to create a road network. Knowledge about roads pertaining specifically to suburban areas is used in the entire extraction process. In this way, the characteristics of suburban areas are considered, for example the fact that road markings are relatively rare in suburban areas, as opposed to inner city areas. Digital surface models are used as additional information, and context objects are extracted in addition to roads to facilitate the selection of the correct roads.

The knowledge-based approach consists of several consecutive steps, starting with a segmentation. In each step, objects are grouped or selected based on a combination of radiometric and geometric features. In the first steps, the radiometric features are the most important features, whereas in later steps the geometric features become more relevant. The initial segmentation is performed using the normalized cuts algorithm, a graph-based algorithm which allows to incorporate information about the desired objects into the segmentation. Another advantage of the normalized cuts algorithm is the inclusion of global properties of an image, thus the algorithm is able to produce segments with smooth boundaries despite disturbances in the object surface. The initial segmentation is followed by a grouping of the segments in order to compensate for oversegmentation. From the grouped segments road parts are extracted. A road part often does not cover a road in its entirety from junction to junction due to disturbances in the road surface or due to other objects which occlude the road. Therefore, extracted road parts which are likely to belong to the same road are connected to subgraphs in the next step. The subgraphs can contain branches which represent several possible courses of the road. These conflicting courses are caused by the presence of falsely extracted road parts. In order to resolve the branches, the subgraphs are evaluated to eliminate those connections which are most likely to be false. The geometric relations between connected road parts are used for the evaluation, as well as context objects which are found in and around the gaps between connected road parts. Context objects are objects which can be found in the vicinity of roads. Some types of context objects, such as vehicles, give supporting evidence for a road hypothesis in the gap between two road parts. Other types of context objects, such as buildings, contradict a road hypothesis if they are found in the gap. After the evaluation and adjustment of the subgraphs, a road network is generated. For this purpose, the roads are represented by approximated centre lines. The network is generated by searching for junctions at the ends of roads. Roads which can be assumed to be wrongly extracted, i.e. short roads that are isolated or parallel and close to longer roads are eliminated. The final road network consists of lines representing the road centre lines and points representing the junctions.

Results are presented for two different data sets. The data sets consist of aerial orthoimages which show suburban scenes and corresponding digital surface models. The results are analysed quantitatively using a set of measures pertaining to the quality of the road extraction, such as the completeness and the correctness, and the quality of the network topology, such as the topological completeness and correctness. The impact of some of the features used in the extraction is tested by performing the extraction without these features and comparing the results to the original results. The results show that the approach is suitable for the extraction of roads in suburban areas.

Keywords: automatic image analysis, road extraction, suburban areas

Zusammenfassung

In dieser Arbeit wird eine neue Methode zur Extraktion von Straßennetzen in Vorstadtgebieten aus optischen Luftbildern entwickelt. Die Straßenextraktionsmethode ist regionenbasiert; Straßenregionen werden aus einem segmentierten Bild extrahiert und miteinander zu einem Straßennetz verbunden. Wissen über die Eigenschaften von Straßen, besonders in Vorstadtgebieten, wird im gesamten Extraktionsprozess genutzt. Auf diese Weise werden die Besonderheiten von Vorstadtgebieten berücksichtigt, zum Beispiel dass Straßenmarkierungen in Vorstadtgebieten relativ selten sind, im Gegensatz zu Innenstadtgebieten. Digitale Oberflächenmodelle werden als zusätzliche Informationsquelle genutzt, und Kontextobjekte werden zusätzlich zu den Straßen extrahiert, um die Auswahl der korrekten Straßen zu vereinfachen.

Der wissensbasierte Ansatz besteht aus mehreren Schritten, angefangen mit einer Segmentierung. In jedem Schritt werden Objekte anhand einer Kombination von radiometrischen und geometrischen Merkmalen gruppiert oder ausgewählt. Die radiometrischen Merkmale überwiegen in den ersten Schritten, während in späteren Schritten die geometrischen Merkmale an Relevanz gewinnen. Die Segmentierung wird mit dem Normalized-Cuts-Algorithmus durchgeführt, einem graphbasierten Algorithmus, mit dem Wissen über die gewünschten Objekte in die Segmentierung integriert werden kann. Ein weiterer Vorteil des Normalized-Cuts-Algorithmus ist die Einbeziehung globaler Bildeigenschaften, so dass der Algorithmus trotz Störungen in der Objektoberfläche gleichmäßige Segmente erzeugen kann. Nach der Segmentierung werden die Segmente gruppiert, um die Effekte der Übersegmentierung zu beseitigen. Dann werden Straßenstücke aus den gruppierten Segmenten extrahiert. Aufgrund von Störungen in der Straßenoberfläche oder aufgrund von Verdeckungen wird eine Straße häufig nicht vollständig von Kreuzung zu Kreuzung von einem einzigen Straßenstück abgedeckt. Daher werden im nächsten Schritt Straßenstücke, die wahrscheinlich zur gleichen Straße gehören, zu Teilgraphen verbunden. Die Teilgraphen können Verzweigungen enthalten, die mehrere mögliche Straßenverläufe repräsentieren. Diese widersprüchlichen Verläufe entstehen durch die Existenz von falsch extrahierten Straßenstücken. Um die Verzweigungen aufzulösen, werden die Verbindungen in den Teilgraphen bewertet, und Verbindungen, deren Bewertung darauf schließen lässt, dass sie falsch sind, werden entfernt. Für die Bewertung werden geometrische Beziehungen zwischen den verbundenen Straßenstücken und Kontextobjekte in den Lücken zwischen den Straßenstücken benutzt. Kontextobjekte sind Objekte, die in der Umgebung von Straßen gefunden werden können. Einige Kontextobjekte, zum Beispiel Fahrzeuge, bieten unterstützende Hinweise für Straßenhypothesen in Lücken zwischen zwei Straßenstücken. Andere Kontextobjekte, zum Beispiel Gebäude, widersprechen einer Straßenhypothese, wenn sie sich in der Lücke befinden. Nach der Bewertung und Anpassung der Teilgraphen wird ein Straßennetz generiert. Dazu werden die Straßen durch approximierte Mittellinien repräsentiert. Das Netz wird durch die Suche nach Kreuzungen an den Enden der Straßen generiert. Straßen, die wahrscheinlich fälschlicherweise extrahiert wurden, vor allem kurze Straßen, die isoliert sind oder parallel zu anderen Straßen mit kurzem Abstand, werden entfernt. Am Ende des Prozesses besteht das extrahierte Straßennetz aus Linien, die die Straßenmittellinien repräsentieren, und Punkten, die die Kreuzungen repräsentieren.

Ergebnisse für zwei verschiedene Datensätze werden vorgestellt. Die Datensätze bestehen aus orthorektifizierten Luftbildern, die Szenen aus Vorstadtgebieten zeigen, und dazugehörigen digitalen Oberflächenmodellen. Die Ergebnisse werden mit Hilfe von Qualitätsmaßen bezogen auf die Straßenextraktion (z. B. Vollständigkeit und Korrektheit) und die Topologie des Netzwerks (z. B. topologische Vollständigkeit und Korrektheit) quantitativ analysiert. Der Einfluss einiger für die Extraktion genutzten Merkmale wird getestet, indem die Extraktion ohne diese Merkmale durchgeführt wird und die Ergebnisse mit den ursprünglichen Ergebnissen verglichen werden. Die Ergebnisse zeigen, dass der Ansatz für die Extraktion von Straßen in Vorstadtgebieten geeignet ist.

Schlagwörter: Automatische Bildanalyse, Straßenextraktion, Vorstadtgebiete

Table of Contents

1 Introduction

In this thesis, a method for the automatic extraction of roads in suburban areas from aerial images is developed. The main goal is to extract a network of the road centre lines.

The road network is an essential part of our infrastructure; roads connect places which are not connected by other means of transport such as railways and planes. Most buildings in Europe are connected to the road network. According to the ERF (European Union Road Federation), 72 % of all inland goods transports and 83 % of all passenger transports in 2008 used roads (ERF 2010). Accurate and up-to-date road databases are very important for using the road infrastructure. The use of road databases for navigation and fleet management is immediately obvious, but they also provide important information for other applications such as traffic monitoring, spatial planning tasks and spatial analysis for a wide range of applications. In (Frizzelle et al., 2009), for example, the authors stress the importance of accurate and complete road data for the analysis of environmental influences on public health. All these applications depend on the existence and quality of the underlying road data.

Road databases must be checked and updated frequently in order to provide accurate and complete data. A common method for the acquisition of road data is the extraction of roads from aerial or satellite images, alone or in combination with other data sources. This work is for the most part done manually, but it is desired to automate it as far as possible in order to save costs and time. For open landscapes, several fairly reliable algorithms for the automatic extraction of roads already exist; a comparison of seven algorithms in an EuroSDR (European Spatial Data Research) test shows that most of the tested algorithms give practically useful results for rural areas (Mayer et al., 2006). However, none of these algorithms gives useful results for urban or even suburban areas. In urban areas, the task of road extraction is more difficult than in rural areas because the environment is far more complex. Roads in urban areas do not stand out against the background as distinctively as in rural areas, so other methods are needed to extract roads in urban areas.

The task of automatic road extraction from aerial images is part of the field of automatic object extraction from images. A main concern for all applications in this field is the reliable identification of the objects of interest. For road extraction, we want to be able to extract roads reliably: as many roads as possible should be extracted, but only those that actually exist in the scene which the image shows. It is very hard to develop a general system which would be able to extract roads in all kinds of images. As mentioned above, methods for extraction in rural areas cannot be easily transferred to urban areas: we need different methods for different environments. Most road extraction methods employ either a line-based road model or a region-based road model; for rural areas line-based models are frequently employed. In urban areas road extraction is generally more difficult than in rural areas because the scene contains more different objects, making it more complex. Under these conditions roads are not easy to recognize as linear objects as in rural areas, so region-based extraction methods with images of higher resolution (1 m or less) are often used in urban areas. Still, as the shape and surface of structures that are not roads can be similar to roads, roads cannot be distinguished from other structures by a single feature. In order to enhance the extraction, a combination of several features, prior information about the road network or additional data sources can be used. In high resolution images, road extraction is sensitive to disturbances by other objects present in the scene which may occlude a road or affect its appearance. If these other objects (context objects) can be extracted and considered in relation to roads, the road extraction can be improved. Globally, the roads form a network whose function is to provide connections between places. In rural areas, the emphasis lies on fast

connections, which is why a condition of following the shortest path between two points can be exploited to optimise the road network. In urban areas the fast connections are not as important, so this condition is much less effective there.

The objective of this thesis is the extraction of road networks in suburban areas. Scenes depicting suburban areas are of medium complexity: they are less complex than inner city areas with their dense development, but still significantly more complex than rural areas. In order to deal with the complexity, knowledge about roads, their appearance and the relations to their surroundings is used, all specific to suburban scenes. New development areas often have the characteristics of suburban areas, and they contain a whole network of new roads, which must be added to a road database. Therefore, no prior database information from road databases is used in the method developed here, in order to be able to extract new road networks. While the road extraction strategy for suburban scenes will be more similar to those employed in urban areas than to those in rural areas, some characteristics typical for suburban scenes need to be considered. Some approaches for road extraction in inner city areas rely, at least partially, on road markings, but in suburban areas roads with road markings are relatively rare; they may even be missing on junctions. Some approaches rely on a regular road grid, which can lead to errors when dead ends are frequent.

The new approach presented in this thesis is tailored specifically to suburban areas. It does not rely on road markings or specific assumptions about the road network. A region-based road extraction strategy is employed, using high resolution aerial images (0.1 m resolution). In contrast to other region-based approaches in urban areas, regions are derived from the image neither by compositions of extracted edges, nor by a supervised classification, but by a segmentation, from which road regions are selected. Knowledge about roads and the various features that distinguish them from their surroundings are employed from the beginning. Additional information in the form of a digital surface model is integrated, and context objects are extracted to aid the road extraction, but both in a manner which does not treat them as essential: if they are not available, roads can still be extracted.

The thesis is structured as follows: In Chapter 2, a literature overview on methods for road extraction is presented. Methods for line-based and region-based extraction are discussed, both for urban and rural areas. In Chapter 3, some techniques that are employed in the road extraction strategy are introduced, namely normalized cuts for segmentation and linear programming for optimisation. In Chapter 4, the road extraction strategy is presented. All steps for the road extraction are described in detail, from the segmentation to the network generation. In Chapter 5, the strategy is applied to some suburban scenes, and a quantitative analysis of the results is presented, as well as an examination of the importance of some parameters. In Chapter 6, the results are discussed and compared to those of other approaches, and some suggestions for improvement of the approach are given.

2 State of the Art

Automatic road extraction belongs to the field of image analysis which deals with automatic object extraction from images. In the following survey, the focus lies on road extraction strategies from remotely sensed optical images from airborne or spaceborne platforms. Their goals may be the automatic extraction of an entire road network, or the verification of road databases. There are many different approaches, using different image sources, images having different resolutions, and following different strategies.

Experiments with automatic road extraction from aerial or satellite images started well over 30 years ago. Compared to today's possibilities, these early works were still to a great degree restricted by the low computational power available and the limited quality of early satellite images and scanned aerial images. However, they laid the foundations for the sophisticated road extraction algorithms developed later; many general strategies originate from the early works, as well as ideas about the main characteristics of road models. One of the earliest approaches in the literature is (Bajcsy and Tavakoli, 1976). The authors extract roads from Landsat-1 images. Of course, due to the very low resolution (80 m ground sampling distance (GSD)) only the most dominant roads like highways could be found at all. Another early work is (Fischler et al., 1981), in which the benefits of using several complementary sources of information from the image are demonstrated. In this work, the need for different strategies for different resolutions is already noted as well as the need for different strategies for rural and urban areas. The approach in (Fischler et al., 1981) was intended for low resolution images in rural areas.

Automatic road extraction approaches are often tailored to a specific type of environment, which is one reason for their great variety. There are approaches for open rural landscapes dominated by open fields, approaches for inner city areas and approaches for semi-urban areas with low-density development; some approaches presuppose a very regular road grid as it is typical for cities in the USA (e.g. Price, 1999). Forested areas are usually not covered by road extraction algorithms based on optical remotely sensed images, because roads in forests are hardly visible even to a human operator. Many automatic approaches are predominantly experimental, but in recent years, automatic approaches for rural areas have become more and more operational. Automatic methods can be roughly classified as either line-based (roads are extracted as lines) or region-based (roads are extracted as elongated regions), though many use elements of both strategies.

The employed road models and extraction strategies depend on the image resolution as well as the general scene content (also called *global context*). Road extraction is performed from aerial and satellite images with resolutions ranging from 80 m to 0.1 m (e.g. Hinz, 2004). In low to medium resolution images (typically in a range between 1 m and 30 m), roads are usually modelled as long thin lines. This is especially true for rural areas, because here roads are often bordered by fields, which appear as relatively large but compact regions in images. Forming a network of long, interconnected lines, roads are distinctly different from most other objects in this resolution and in this global context. In high resolution images (better than 1 m), roads are usually modelled as elongated homogeneous regions. For urban areas, this model is generally preferred because the linear characteristics of roads are not very salient in urban areas. Intersections, where the linear model does not apply, are more frequent, and there are many other linear features, mainly from buildings.

In many approaches, road extraction is subdivided into two steps: a local step where single road segments are extracted based on local characteristics of roads, and a global step, where

the extracted roads are grouped and connected to form a network (e.g. Wiedemann, 2002). Grouping involves the bridging of gaps and the elimination of false extractions. Region-based and line-based approaches mostly differ in the way the local analysis is carried out, whereas the methods applied for grouping are often similar. Some form of junction extraction is necessary in the grouping step, whether implicitly by joining several road segments or explicitly based on a separate junction model. Some approaches, especially experimental ones, do not perform the global step but terminate after a local road extraction.

After the road extraction, the extracted roads are in general represented in one of two forms: as regions representing the road surface, or as lines representing the road centre line. In line-based approaches, the representation is naturally the latter. In region-based approaches, the centre line must be derived from the road region if a representation by centre line is desired. Often, a representation by centre line, possibly with attributes such as the associated width of the road, the number of lanes, etc., is a more useful representation for subsequent applications.

Several kinds of additional data sources can be exploited for road extraction in addition to the images. These include geo-spatial databases containing road data or digital surface models (DSM), the latter derived, for instance, from image matching or from LIDAR (light detection and ranging) data (e.g. Hu et al. 2004b). Prior information about the structure of the road network is another source of additional information, which can be used in terms of constraints on extracted elements and their relations. This is sometimes used for road extraction in certain types of urban areas where the road network consists of a regular grid composed of straight roads (e.g. Youn et al., 2008). Additional information can also be acquired directly from the image by extracting other objects than roads. These so-called *context objects* and are useful to consider because roads are not isolated objects; their appearance in images is often affected by other objects that are close to them or even occlude them, such as vehicles, trees or buildings. Therefore, some approaches (e.g. Zhang, 2004) do not only extract roads but also one or more types of context objects. Frequently, these context objects are used in order to decide whether a gap between two road segments, extracted in the first stage of road extraction, can be bridged or not.

A frequent application for road extraction algorithms is to enhance the quality of existing road databases (e.g. Ziems, 2010). In this case, information from the database is used in various extents to aid the road extraction. The exploitation of database information ranges from a complete adoption of the topology via the determination of regions of interest or road extraction parameters to road extraction independent from the database (i.e. in the latter case, the database is not used in the extraction process, only for change detection).

In addition to automatic road extraction approaches, there is also a variety of semi-automatic approaches, where user input is required to indicate some parts of the roads. The algorithm searches for the correct course of the road starting from the parts indicated by the user. In general, semi-automatic approaches, also called road tracking approaches, focus on immediate practical application, whereas many fully automatic approaches have a more experimental focus.

This literature overview starts with an overview on semi-automatic approaches in Section 2.1. Line-based approaches for road extraction are covered in Section 2.2, and region-based approaches are described in Section 2.3. Examples for the integration of additional information such as road databases, DSMs or extracted context objects are given in Section 2.4. In Section 2.5, the extraction of junctions is discussed, and in Section 2.6, a summary of the literature review is given.

2.1 Semi-automatic Methods

The main idea of semi-automatic methods is to give the algorithm one or more starting points that are guaranteed to lie on the road. This is usually done by indicating the starting points manually. The tracking algorithm then tries to extract the course of the road by predicting the position of the next road point and improving this position using image information. One of the earliest tracking approaches on high resolution images (30 cm to 1 m) can be found in (Quam, 1978), where the intensity cross profile of the already extracted road is matched with the cross profile at the predicted continuation in order to track the road.

Many road trackers (e.g. McKeown and Denlinger, 1988; Vosselman and de Knecht, 1995) start with one seed point per road, a corresponding start direction (usually given by indicating a second point) and possibly a start width. The tracking iterations usually consist of a prediction step and an update step. In the prediction step, the next road point is predicted from the course of the road up to this point, but without using image information, for example by parabola fitting (McKeown and Denlinger, 1988; Lin et al., 2008) or by applying a Kalman filter to predict position, direction and direction change (Vosselman and de Knecht, 1995; Zhou et al., 2006). Zhou et al. (2006) compared the performance of the Kalman filter with that of a particle filter; they found them similar with the particle filter being slightly superior. All approaches use high to medium resolution images (between 0.6 m and 1.6 m GSD), McKeown and Denlinger (1988) also use images with a resolution of 3.5 m.

In the update step, image content is used to decide whether evidence for a road exists for the predicted position or if the road position must be shifted. Cross section profiles of the road are often used for this purpose (Vosselman and de Knecht, 1995; Zhou et al., 2006). The cross section at the predicted position is compared to a model cross section obtained either from the starting point alone or from an average of the past road positions. McKeown and Denlinger (1988) combine the surface-based cross-section tracker of Quam (1978) with a method for antiparallel edge detection developed by Nevatia and Babu (1980); edges are extracted parallel to the predicted road direction. Both trackers support each other: if one fails, it can be restarted from the other if that one was successful. In this way the combined tracker can get over places where the road information is partly corrupted. Another possibility to obtain more robust image information is to use a criterion which is less local. Lin et al. (2008) use Angular Texture Signature (Haverkamp, 2002) to find the new road position and direction. In essence, they measure the mean or standard deviation of grey values of a rectangular patch in several directions around the predicted road direction.

Another way for integrating manual input into road extraction is that the human operator sets several seed points to indicate the approximate course of the road. The task of the algorithm is the optimisation of the course of the road between these points. Here, practical application is emphasised over extensive automation: the main task is to assist the human operator to extract the road accurately. One example for this class of tracking methods is Hu et al. (2004a), who find the road position between two points by parabola fitting using edge information from the image. The road pieces are consecutively fitted following the user input. Another approach that uses seed points along roads is (Grün and Li, 1995). This approach uses an image derived from the original image where edge information is enhanced by a wavelet transform. Seed points are connected by optimisation of the course between them. The optimisation is performed using snakes (Kass et al., 1988), a form of active contours.

Hu et al. (2007) track roads using so-called footprints of pixels. A footprint is a homogeneous region around a pixel. The roads are extracted by tracking in the directions of the “toes” of the

footprints, which are long branches going out from the centre; new road positions are located in the centres of the toes. Unlike most other tracking algorithms, this approach allows the identification of junctions by their footprints, which have more than two toes, and by following all identified toes at intersections, a network can be extracted. The algorithm relies on a very homogeneous road surface with clearly visible road sides; disturbances such as cars on the road can cause the algorithm to fail. Seed points for the start footprints are set manually, but can also be selected automatically as footprints that are rectangular and elongated; in this case the user must specify whether the road is brighter or darker than the surroundings. If the roads do not have a significant contrast to the surroundings, the automatic seeding is likely to fail.

To summarise, most tracking approaches circumvent the problem of detecting roads in aerial and satellite images by using seed points provided by a human operator for the extraction. The subsequent tracking is sufficiently accurate to delineate the road centre lines for practical applications such as road database update. Since the existence of the road is already established by the user input, small gaps can be bridged quite reliably. Thus, tracking algorithms can be used as an efficient assistance for a human operator for the purpose of road extraction.

The main disadvantage of tracking algorithms is that they rely on accurate seed points, which must be provided manually in most cases. Most algorithms use only very local image information and knowledge about roads; they cannot extract a whole network (with the exception of Hu et al. (2007), who use an extensive pruning algorithm after the tracking in order to eliminate false branches). Tracking algorithms which use one seed point from which the course of the road is detected cannot deal with disturbances of road information larger than a few steps: they cannot decide where the road tracking can be resumed, requiring a manual restart. Algorithms which seek to optimise the course of the road between seed points, on the other hand, do not encounter this problem, but they rely heavily on the correct input topology: a road delineation is delivered whether the road actually exists or not.

2.2 Line-Based Road Extraction Methods

The road model used for many line-based approaches is similar to the one described in (Wiedemann, 2002): roads are modelled as long lines with a certain width of a few pixels (depending on the road class and the image resolution), with relatively homogeneous grey values that are either darker or brighter than the surroundings and have a more or less significant contrast to the areas at both sides. On a more global level, they have limited curvature and are connected to form a road network. Most approaches first extract linear features locally and afterwards group them to roads and road networks.

Several approaches, all developed for rural areas (Baumgartner et al., 1999; Wiedemann, 2002; Bacher and Mayer, 2005; Gerke, 2006; Ziems et al., 2007), apply the line extraction algorithm developed by Steger (1998). With this algorithm, lines within a certain width range can be extracted and represented by their centre lines and widths. The results are separate line segments which are connected to a road network in a second step. This approach is enhanced in various ways by different authors. For example, line extraction in low resolution images is combined with edge extraction in high resolution images in (Baumgartner et al., 1999; Baumgartner, 2003). Bacher and Mayer (2006) use a two stage process where the results of the first extraction are used as training areas to classify the image; this is followed by a second line extraction process in areas classified as roads. The line extraction in (Steger, 1998) is also used in the context of road database verification, by comparing the extracted roads to the

database roads (Gerke, 2006; Gerke and Heipke, 2008). Ziems et al. (2007) additionally use database information to adjust parameters for a pre-segmentation and for the line extraction.

After the extraction of single line segments, these segments have to be grouped into complete roads and road networks. Baumgartner et al. (1999) group road segments iteratively, using criteria such as distance and collinearity to establish a connection. Connecting lines are optimised by ribbon snakes (Fua, 1996), where in a first step the position of the centre line is found and in a second step the line width is optimised. To accept a connection, the width variation must be low. If a connection could not be verified by grey values or snakes, local context objects (buildings, shadows) are searched and evaluated. Wiedemann and Ebner (2000) use a similar approach but add a follow-up step to complete the road network considering the quality of the network's connection function. Bacher and Mayer (2005) and Gerke (2006) also use this strategy; Gerke (2006) extends it by using a road database as additional information and rows of trees as context objects.

In some of the earlier approaches on satellite images (Wang and Newkirk, 1988; Ton et al., 1989), major roads are extracted from Landsat TM images (30 m resolution) by a strategy where first potential road line pixels are extracted, which are then grouped locally to line segments and afterwards grouped to longer collinear straight lines across gaps. Other authors (e.g. Fischler et al., 1981; Klang, 1998) pre-process the images such that pixels that are more likely to belong to lines have higher grey values, and proceed to connect a road network directly on these images. Fischler et al. (1981) use and combine several operators that classify pixels with high grey values as road pixels, some with high correctness and some with high completeness. Klang (1998) calculates the eigenvalues of the Hessian matrix for each pixel. In approaches whose first step is the calculation of an enhanced image the second step is usually a path search. In (Fischler et al., 1981) this is done as an optimisation seeking the path with minimum cost through the image. In (Klang, 1998), lines are tracked to update an existing road database, starting from seed points which are taken from the road database.

There are some line-based approaches that were developed specifically for urban areas. Because of the complexity of the scene (which contains many objects with linear features besides roads) additional prior knowledge, constraints or data sources are employed. A frequent assumption is a regular grid of straight streets (Youn et al., 2008; Hu et al., 2004b); consequently, these approaches attempt to find long straight lines as roads. Youn et al. (2008) first partition the image such that each part has only two dominant line directions and then locate the road centre lines where a long line intersects with only few other line pixels and runs through pairs of parallel lines. Hu et al. (2004b) determine regions of interest from a combination of a LIDAR DSM and a vegetation classification, and find long lines in the regions of interest by a Hough transform. An obvious disadvantage of these methods which are based on straight lines is that they cannot deal with curved roads. Shackelford and Davis (2003) also extract straight lines by determining for each pixel the longest line of spectrally similar pixels after having excluded vegetated regions. Road segment candidates are long lines where the directions of most lines through the pixels are similar. The subsequent grouping allows roads having limited curvature, controlled by the minimum allowed segment length and the direction similarity. Another approach for urban areas where extracted lines are grouped is (Hu and Tao, 2007). The authors limit the extraction objective to major roads which usually have distinctive linear characteristics even in urban areas. Extracted lines are grouped hierarchically, starting with the longest and straightest lines.

Some approaches use energy models of road networks. Stoica et al. (2004) use statistical sampling for road network extraction. They employ a Gibbs point process (van Lieshout,

2000) where interconnected road segments are created, changed or deleted in a stochastically controlled process. The process can start with a random initialisation; in the experiments the process was started without any segments. In each step, one segment can be created, changed or deleted. An energy function whose global minimum indicates the optimal road network is calculated from an interaction term (connectivity of road network) and a data term (segments with grey value features typical of lines are favoured). The steps are controlled by a Reversible Jump Markov Chain Monte Carlo procedure (Green, 1995) in conjunction with simulated annealing in order to avoid terminating at local minima of the energy function. Working with very low resolution images (25 m), Géraud and Mouret (2004) first perform a watershed segmentation on a potential image (an image where pixels likely to belong to a road have a high grey value). The watersheds are assumed to contain the road network. The road network is then optimised using a Markov random field and simulated annealing.

Line-based extraction methods yield good results in open rural areas; some algorithms developed for these areas are already operational. However, line-based methods are not suited very well for urban areas. If line-based methods are employed in urban areas, they are usually limited to the extraction of long straight roads (e.g. Youn et al., 2008) or major roads (Hu and Tao, 2007). This restriction is necessary because of the multitude of linear features that can be observed in urban areas, and severely limits the scope of line-based road extraction in urban areas.

2.3 Region-Based Road Extraction Methods

For region-based road extraction algorithms, roads are modelled as elongated regions with a limited range of widths. Region-based algorithms are usually developed for images with resolutions of 1 m or higher (aerial images or IKONOS or Quickbird satellite images). In very high resolution, most approaches are region-based. Roads are often modelled segment-wise as elongated, more or less rectangular regions with a homogeneous grey value distribution. The colour of the road region lies within a certain range of colours, though this range is highly dependent on the area, the sensor, and the illumination. Some algorithms explicitly use the road borders, modelled as two parallel lines, which enclose the road region. Problems with the extraction can occur where these assumptions are not met, for example where other objects such as cars break the homogeneity of the surface, or where parts of the surface are occluded by trees. These kinds of disturbances have higher impact in high resolution images.

There is a great variety of region-based road extraction methods. Road segments can be composed from extracted parallel edges (e.g. Hinz, 2004; Hinz and Baumgartner, 2003) or directly extracted as regions, either from a multispectral classification (e.g. Zhang and Couloigner, 2006) or from a segmentation (Ruskoné and Airault, 1997). Some approaches use a multi-spectral classification to determine regions of interest in order to limit the road extraction to certain areas (e.g. Zhang, 2004).

There are several different approaches using multi-spectral classification for road extraction in rural areas. Zhang (2004) constructs road segments from parallel edges. The goal is the geometrical improvement of roads in a database. Regions of interest around the database roads are first determined by an unsupervised classification (ISODATA; Jain and Dubes, 1988) with three bands derived from the original RGB bands: the first band of the principal component transform, a greenness band and the saturation of the HSI colour space. The road class, combined with low regions from a DSM, defines the region of interest, in which parallel edges are considered to delimit road segments. It is not clear how the road class is identified among the classes from the unsupervised classification. The evaluation of various

extracted features, including road markings and zebra crossings, is guided by an extensive set of rules. In order to build the road network, Zhang (2004) first closes small gaps by bridging them directly and then builds the network through optimisation, taking gap lengths and the existing road database into account.

Mena and Malpica (2005) use three different supervised classification methods: a pixel-based multi-spectral classification, a comparison of the colour distribution of each pixel's neighbourhood with the distribution of the training samples, and a classification based on Haralick features of the co-occurrence matrix. The training areas are derived from GIS roads. The results of all three classification methods are fused using an evidence-based approach, yielding the road regions. Doucette et al. (2001) use hyper-spectral data (210 bands) in a supervised classification and afterwards group road pixels with a k-median clustering. Then they link clusters iteratively using a self organising map; isolated clusters are discarded. Mohammadzadeh et al. (2006) extract road regions by a classification based on fuzzy logic. The road segments are enhanced by a sequence of morphological operations such as opening, closing and the selection of elongated segments, in order to remove small extrusions and fill holes. All three approaches rely on roads having distinct radiometric characteristics; roads cannot be extracted where their surface deviates too much from that of the training samples.

Zhang and Couloigner (2006) employ a multi-spectral classification for road extraction in urban areas. They perform an unsupervised k-means classification on IKONOS images (four bands) as a first step. The road class is identified using a fuzzy classifier based on the assumption that for roads the reflectivity in the infrared band is lower than in the other bands. The pixels belonging to the road class are further classified using a shape-based descriptor (Angular Texture Signature; cf. Haverkamp, 2002; Gibson, 2003); only pixels that belong to elongated regions are kept. Many false extractions occur where regions have similar spectral characteristics to roads. Haverkamp (2002) also performs a multi-spectral classification and then extracts pixels based on the shape of the regions they lie in. Unlike Zhang and Couloigner (2006), Haverkamp (2002) assumes a straight road grid. Ma et al. (2008) also use multi-spectral classification to identify potential road regions, followed by a second shape-based classification. They use support vector machines (SVM) for the multi-spectral classification. For the shape-based classification, the two main edge directions in the image are determined, assuming a straight regular road grid. Pixels located in a road region are extracted as road regions if there are enough other road class pixels in at least one of two elongated templates oriented along the main directions.

Poullis and You (2010) use combined colour and orientation information to classify high-resolution satellite images in urban areas into road pixels and non-road pixels. For each pixel they determine if it lies on a curve, and, if it does, the orientation of the curve at this pixel, and use this information in addition to foreground and background colour information (provided by training areas) in a graph cut algorithm (Boykov et al., 2001). Road centre lines are extracted with an iterative Hough transform. The centre lines are combined to a road network by an extensive interactive editing.

Hinz (2004) extracts roads in dense urban areas based on extracted edges in multiple view grey value images. Bright lines (assumed to be road markings) and edges (assumed to be roadsides) in high resolution images (0.2 m) as well as ribbons in a lower resolution are extracted within regions of interest determined from a DSM. The extracted lines, edges and ribbons are then combined to road lanes. Collinear lanes and parallel lanes are grouped to road segments, and road segments from different images are fused. Finally, the road segments are connected to a network. The extraction is guided by several rules and internal confidence

checks using fuzzy membership functions to compare features of the extracted object to modelled features. The approach is designed for inner city areas: it requires a DSM and assumes buildings to be present, as well as road markings. The extracted road segments are grouped by looking for shortest paths in a weighted graph. Dal Poz et al. (2006) also extract road regions based on edges; they extract edge polygons and combine them to roadsides. Their method was developed for rural areas.

A rarely used possibility to obtain road segments is to extract them from a segmented image. Ruskoné and Airault (1997) perform a watershed segmentation (Beucher and Meyer, 1993) as a first step but they do not use the obtained segments directly. The borders of the segments (the watersheds) are approximated by polygons. Then parallel edges are searched for among the segment borders. Elongated regions are constructed from the extracted parallel edges; they are used as seed road segments. The road segments are expanded along their main direction according to a homogeneity criterion. Afterwards, the extracted road segments are connected and checked for consistency in an iterative process. The approach is intended mainly for rural or at most semi-urban areas.

Peng et al. (2008) describe an approach for dense urban areas which uses a global energy minimisation method for a higher-order active contour model expressed as a phase field (Rochery et al., 2005). This model is a variant of the level set method (Malladi et al., 1995). A region in the image is described by an energy function, consisting of data energy and geometric energy, which is to be minimised. In contrast to the original level set method, the region can be initialised arbitrarily and is iteratively optimised by a gradient descent method. After the optimisation, the region which minimises the energy function describes the road network. The data energy is derived from the image intensity using Gaussian mixture models, which are learned in a supervised way from samples of the road and background classes. The geometric energy, which is the key factor in this approach, favours a region with smooth borders and elongated arms within the range of expected road widths. With a local energy term the region is smoothed, a non-local term prevents merging of opposite road sides, and another non-local term favours elongated regions. Only major roads can be extracted, and the extraction fails where the intensity of the road surface does not correspond to the learned intensity model.

One aspect that has to be considered especially in region-based extraction approaches is the centre line extraction. The road centre line is needed for many applications, but region-based approaches first and foremost deliver the extracted roads as regions. The centre line extraction is not trivial if roads are extracted directly as regions, because the road borders can have irregular shapes and extrusions into adjacent pavements, parking areas, driveways etc. In several approaches the road centre line is defined as the skeleton, for example in Mohammadzadeh et al. (2006). This skeleton is often determined using morphological operations (e.g. Soille, 1999). Mena and Malpica (2005) extract two skeletons after smoothing the edges of the road regions: a detailed skeleton, which contains all extracted roads but is not necessarily smooth, and a coarse skeleton, which consists of straight lines, but may not be geometrically accurate and miss some smaller roads. Zhang and Couloigner (2006) extract road centre lines with a localised Radon transform. Clode et al. (2007), who extract roads by classification of a LIDAR point cloud, vectorise the road regions with a so called Phase Coded Disk (PCD). The PCD, defined by a complex exponential function, is convolved with the road regions. The result consists of an amplitude image and a phase image; the location of the centre line as well as its width can be derived from the amplitude image, and the direction of the road can be derived from the phase image.

Some approaches combine region-based and line-based methods. The road extraction itself is line-based, but it is enhanced by a multispectral classification, in order to make better use of the colour information provided by multi-channel images. The classification is supported by information about the roads obtained previously. Bacher and Mayer (2005), for example, employ the line-based extraction developed by Steger (1998) on all channels of a multispectral image. Training areas for a multispectral classification are obtained from the extracted lines; the fuzzy classification yields a membership value for the road class for each pixel, leading to a road class image where pixels with high membership values for the road class have higher grey values. The line extraction is performed in the road class image, and results from the original channels and the road class image are fused. The obtained road segments are then combined to a network by closing gaps if they are part of an optimal, i.e. shortest, path between road points. Short gaps are closed immediately, whereas connection hypotheses for larger gaps first have to be optimised using ziplock snakes. They are evaluated using the grey value profile along the verified path.

A similar strategy is employed by Doucette et al. (2004). As in the previous example, they start with a preliminary road extraction and use the preliminary extracted roads to generate training samples for a supervised classification. Afterwards, road extraction is repeated only within the road class. Road centre line pixels are extracted as centre pixels between anti-parallel edge pixels; a road can be found where enough parallel edge segments are visible. The extracted road centre line pixels are iteratively grouped into clusters which represent short line segments. The line segments are grouped to a road network using a weighted graph. Areas around the nodes from the road network provide training samples for the multi-spectral classification which is done with the maximum likelihood method. After the classification, the process of extracting and grouping centre line pixels is repeated within the regions classified as road.

Ziems et al. (2007) also use a multi-spectral classification to enhance the road extraction process, but here GIS roads are used to obtain training samples instead of extracted roads, as the aim is a verification of the roads in the GIS. Radiometric parameters in the locations of the GIS roads are used to classify all pixels into road pixels and non-road pixels. The colour information from the locations of GIS roads is additionally used to tune the parameters for the subsequent line-based road extraction, which is carried out using the method of Wiedemann (2002).

As the above survey shows, region-based algorithms for road extraction exist in a great variety, both for rural and urban areas. The region-based algorithms for high resolution images are better suited for urban areas than line-based algorithms because in the region-based extraction, more features can be considered. This facilitates extraction in complex urban surroundings. One problem of region-based algorithms, which is aggravated in higher resolution images, is the difficulty to derive an accurate centre line from the extracted regions. The centre lines are often derived from skeletons, but as the borders of the extracted regions can be quite irregular due to shadows, occlusions and other disturbances of the visible road surface, the skeletons will also have irregular shapes, especially at the ends of roads and at junctions.

2.4 Use of Additional Information

Sources of additional information include existing databases, DSMs and prior information about the structure of the road network. Another source of additional information are context objects detected in the images. Many of the algorithms mentioned in this section were

previously described in Sections 2.2 and 2.3; they are reviewed here with the emphasis on the respective type of additional information.

Several approaches for road extraction in order to verify or enhance road databases rely to some extent on the existing road database. Klang (1998) takes the whole topology of the database network as a starting point for the road extraction. The intersections of the database roads are matched with intersections of lines in the image. The matched intersections serve as starting points for a ziplock-snake algorithm to find the course of the roads. Obviously, the approach relies heavily on the topological correctness of the road database as the snake-based approach will find a course whether a road is visible in the image or not. Göpfert and Rottensteiner (2009) use network snakes, with the data energy taken from LIDAR data, to enhance the 3-dimensional positions of the database roads.

Stilla (1995) uses the information from a map to control the order in which extracted objects are grouped. Compositions of line segments from the existing map are stored in an image description graph. Extracted lines in the image are also grouped iteratively into composite objects to form a road network of lines and crossings. The order in which lines or composite objects are processed in each iteration cycle depends in part on their compliance with the corresponding objects in the map. Objects that correspond closely with the objects in the map have a higher priority.

Ziems et al. (2007) use database information to adjust parameters for a colour segmentation to obtain regions of interest as well as for the line-based road extraction. They are determined from the positions of the database road in the image and the adjacent areas. Mena and Malpica (2005) use colour information from database roads in a similar way as training areas for a multi-spectral classification. The database roads also assist in the last stage in which a topological description of the extracted road network is derived.

Ziems et al. (2010) describe a road database verification approach where they look for supporting evidence for a road at the location of each database road object. They use several modules with different approaches for road extraction, road verification and extraction of other objects. The results of the modules are combined to decide whether the road object is accepted or rejected, or whether the combination of results does not give a clear answer, in which case the user needs to inspect the road object.

Gerke and Heipke (2008) describe an approach for the verification of road databases where the road extraction is at first independent from the database. After the road extraction, the roads in the database are assessed in a two-stage process. In the first stage, it is checked whether the database roads correspond to the extracted roads. Roads which could not be accepted straight away are further inspected by assessing their importance for the network and a second extraction with relaxed parameters.

DSMs can be used as primary data sources for road extraction (e.g. Clode et al., 2007), but they are also often used as additional data source in combination with images. As roads are objects on the ground, a DSM can be effectively used to define regions of interest for the road extraction by excluding regions above ground (Zhang, 2004; Hinz, 2004; Hu et al., 2004b). In order to properly identify regions above ground, a normalised DSM (nDSM) has to be derived from the DSM. In a normalised DSM the height of the ground is zero; only objects above ground have heights different from zero. A nDSM is calculated by subtracting the Digital Terrain Model (DTM), which does not contain objects above ground, from the DSM. If only a DSM is available, the DTM must first be derived from the DSM. This can be done by filtering

the DSM such that objects above ground are eliminated but the height variations of the ground itself are preserved, for example by morphological grey value opening (Weidner and Förstner, 1995). Another use for the DSM is the identification of shadow regions, as described by Hinz (2004): in combination with the image acquisition time, a ray tracing in the DSM predicts the location of shadow areas in the image. This information is used to adapt extraction parameters that depend on the intensity. Hu et al. (2004b) use LIDAR data to obtain height information; additionally, they use the intensity of the LIDAR data in combination with the information of an optical image to classify the image into vegetated and non-vegetated areas.

Prior knowledge about the shape of the road network is sometimes used, most often by assuming that the road network is a regular grid. Line-based extraction strategies in urban areas often employ this type of constraint, for example (Shackelford and Davis, 2003; Youn et al., 2008; Hu et al., 2004b). Shackelford and Davis (2003) extract long lines with little grey value variation. Youn et al. (2008) extract long lines which cross only few other lines. Hu et al. (2004b) extract long lines within the valleys of a DSM using the Hough transform. The limitation to a regular road grid with straight roads is also used by some region-based approaches. Price (1999) is an example for this. The extraction is started manually by indicating three intersections which allow to estimate the two main orientations and the grid spacing. Road segment hypotheses are created between the intersections, and new intersections and segments are iteratively added to the grid. The road sides of the new segment hypotheses are checked against extracted edges and receive a quality measure. Widths and positions of extracted segments are adjusted according to the neighbouring segments. The network is checked and refined according to the quality measures of the road segments. Sequences of segments with poor quality measures can be deleted. Another region-based approach that exploits the grid character of the road network is Haverkamp (2002). These approaches are naturally only suitable for urban areas which correspond to the fairly restrictive network model.

Several approaches use context objects to aid the road extraction. Hinz (2004), for example, extracts vehicles and rows of vehicles. A gap between two lane segments can be closed if a vehicle is found inside the gap. Rows of vehicles are evidence for lanes and are used in the lane fusion process. The vehicles are extracted using a 3D wireframe model. Vehicles as context objects are also used by Hu et al. (2004b) to identify parking lots. They extract vehicles with a pixel-based classification using training areas from manually chosen vehicles.

In the approach of Zhang (2004) several kinds of context objects are used in order to evaluate gaps and missing road sides and to decide whether a gap can be bridged. Buildings, trees and vegetated areas are extracted by combining the DSM with the results of a multi-spectral classification. A gap is accepted as road if trees or shadow areas are found inside the gap; buildings are not allowed inside the gap. Baumgartner et al. (1999) also use several context objects in order to verify connections between extracted road segments. The relations between roads and several context objects are modelled; buildings or trees, for example, can occlude the road or cast shadows. Driveways which branch off from the main roads are also considered as context objects, as well as vehicles. Gerke and Heipke (2008) use rows of trees as additional evidence for roads.

Ruskoné and Airault (1997) use several land cover types to verify extracted road hypotheses. After a preliminary road network is extracted, all road hypotheses are divided into small sections of equal length and for each section the most probable land cover type is determined by a supervised classification. Five land cover types are considered: roads, crossroads, trees,

shadows and fields. The sections are then evaluated using their land cover types, shapes and relations to neighbour segments: they are either accepted or rejected as roads in order to establish the final road network.

Using additional information can facilitate and improve the road extraction; however, it often introduces restrictions as well. Additional data sources such as databases and DSMs can only be employed if they are available at all, and introducing prior knowledge about the road network restricts the types of road networks that can be extracted. If context objects are used, they have to be extracted additionally, which adds to the complexity, or imported from other sources, which have to be available. Still, as sources of additional information can improve the road extraction significantly, their advantages should be exploited. For the purpose of extracting new roads, for which database information does not exist, the use of DSMs and context objects is beneficial. Both are useful to exclude wrong extractions and can help to close gaps between extracted roads.

2.5 Junctions

Junctions are difficult to extract within the process of road extraction because their appearance in the image differs from that of the roads. Many road extraction methods exploit the fact that roads are rather narrow regions bordered by edges. Thus, the road extraction usually breaks down in the locations of junctions. There are different ways of handling this problem. Some approaches do not attempt to extract junctions at all, extracting only roads independent from each other. In other approaches, junctions are modelled implicitly as the points of connections between roads, and some approaches extract explicitly modelled junctions.

Approaches where junctions are not considered at all are often tracking approaches. They simply stop at junctions since the tracking conditions are no longer fulfilled (e.g. Zhou et al., 2006).

In many approaches junctions are not modelled explicitly. Roads are often extracted as independent entities without considering junctions. The intersections of the lines are considered as junctions without any further checks of geometric or topological integrity (e.g. Hu and Tao, 2007; Hu et al., 2004b; Youn et al., 2008). If the road lines could be extracted with good accuracy, this procedure is sufficient to determine the locations of junctions. In region-based approaches the intersections of skeletons can be used to determine junction points (Ma et al., 2008), but often, skeletons are extracted without further identifying junctions (e.g. Mohammadzadeh et al., 2006). Similarly, in the optimisation algorithms of Peng et al. (2008) and Stoica et al. (2004), road networks are detected as a whole without separate detection of junctions.

In some approaches junctions are modelled and extracted explicitly, which is done in various ways. For example, pixels can be classified according to features of their neighbourhood such that junction pixels can be distinguished from ordinary road pixels. Basically, more than two elongated homogeneous regions proceed from junction pixels, contrasted to only two from road pixels (Haverkamp, 2002; Hu et al., 2007). In Hinz (2004), junctions are identified after the road network extraction as network nodes with a degree higher than two. To be accepted as valid junctions, the node degree must not exceed five, because it is very improbable to encounter junctions with more than five roads. The junctions are then classified into simple junctions (which can be extracted as nodes from a line extraction in low resolution) and complex junctions (where the line extraction in low resolution fails). Ruskoné and Airault

(1997) distinguish roads and crossroads in the classification of land cover types which is used to verify extracted segments. Crossroads have similar radiometric characteristics to roads, but their shape is not elongated, and there are several road segments in their vicinity.

In Baumgartner et al. (1997), junction areas are extracted separately as polygons of the road borders after the network construction. Junctions are located at places where extracted centre lines meet at approximately perpendicular angles, and the associated road segments are used to determine a search area for polygon edges. Extracted polygon edges are combined with the ends of the road segments to define the junction area. Wiedemann (2002) also treats junctions separately after the road network extraction is completed. As the network extraction can lead to topological errors at junctions, these are corrected first. Then, the extracted junction is compared to several model junctions representing typical appearances of junctions and the best fitting model is assigned to the junction.

Some approaches even detect junctions first and continue with the road extraction proceeding from the junctions. Klang (1998), whose main objective is to improve the geometric accuracy of a database, starts with extracting junctions from the database and locating their accurate positions in the image. The roads between the junctions are extracted afterwards. In Price (1999) the human operator must first indicate three junctions manually from which road grid properties are derived. The locations of other junctions are predicted and adjusted according to image content.

Several approaches are specifically dedicated to the extraction of junctions. One example is (Ravanbakhsh et al., 2008a), where the approximate location of the junction is derived from a database. Using database information about the number and direction of roads associated to the junction, road arms are built from extracted and grouped edges. The borders of the road arms representing the roadsides are connected, and the outline of the junction is refined using ziplock snakes. The approach is augmented in (Ravanbaksh et al., 2008b) by the additional extraction of traffic islands in complex junctions using level sets, and in (Ravanbakhsh and Fraser, 2009) by the extraction of roundabouts. Another junction extraction algorithm, based on neural networks, is described by Barsi et al. (2002).

Usually, a road network can be extracted without a sophisticated junction model, but the geometric accuracy can be decreased in the junction area (the position of the extracted junction centre deviates from the correct position). This is also true for the topological accuracy. For example, one four-arm-intersection can be confused with two three-arm-intersections close together. In areas with heavy traffic, the junction extraction can be made impossible by rows of vehicles waiting at the junction if vehicles are not considered in the extraction.

2.6 Summary

The goal in this work is the automatic extraction of road networks in suburban areas. From the literature survey it can be concluded that region-based approaches on high resolution images are more suitable for urban and suburban areas than line-based approaches. Line-based approaches often fail in complex urban environments because the linear structure of most roads does not stand out enough among a multitude of other linear features. They can still be used for the extraction of major roads or in combination with geometric constraints, but region-based approaches are more flexible, especially when dealing with curved roads and varying road network patterns. An often employed technique to extract road regions is multi-spectral classification. In order to employ this technique, knowledge about the radiometric

properties of the roads in the image has to be provided by training samples or from already existing database roads; if parts of roads deviate in colour from the learned samples, the road extraction fails. Image segmentation, for which prior knowledge about the road colour is not necessary and which would therefore avoid this problem, is used very rarely. One exception is (Ruskoné and Airault, 1996), which is developed mainly for rural areas.

Road extraction algorithms developed for urban areas usually employ some kind of additional information to overcome the difficulties posed by the complex surroundings. While the use of additional information is almost always beneficial, some kinds of prior information such as assumptions about the shape and pattern of the road network can limit the extraction. The use of existing road databases can facilitate the extraction, but methods that rely strongly on them cannot be used where road data are not available. Suitable sources of additional information that avoid these problems are DSMs because in suburban areas the roads usually lie on the terrain surface. Considering context objects can also help the extraction, both implicitly – by allowing for fragmented road extraction – and explicitly by extracting context objects common to urban areas and analysing their relations to extracted roads or gaps between extracted roads. There are not many approaches for urban areas which explicitly use context objects; when approaches incorporate them, buildings and vehicles are the most commonly used context objects, while vegetation is often used indirectly in a classification. Shadows can be used directly as context objects similar to trees or buildings, or indirectly by modification of extraction parameters in shadowed areas.

Most approaches for urban areas are developed for image resolutions around 1 m; approaches using high resolution (better than 0.5 m GSD) are very rare. One exception is (Hinz, 2004), an approach which relies heavily on road markings. There are no methods for high resolution images that are explicitly developed for suburban areas. High resolution images contain more small objects which can disturb the homogeneity of the road surfaces; on the other hand, they also provide more information than images with lower resolution. Smaller roads in particular are less likely to blend into their surroundings.

Concluding from the findings mentioned above, we aim at a region-based road extraction algorithm which is flexible enough to not rely on assumptions about the road surface colour, thus using segmentation as a first step. A model-based approach is developed, and domain specific knowledge is used already in the segmentation because the segmentation is the foundation for the later steps: its quality greatly influences the quality of the extraction results. Primary data sources are high resolution (0.1 m) colour infrared (CIR) images. DSMs and context objects will be employed as additional information but the algorithm should not rely completely on them in order to remain operable without them.

3 Basics

In this chapter, two methods used for the road extraction that are assumed not to be generally known are explained in detail. In Section 3.1 the normalized cuts method for the segmentation is presented. In Section 3.2 the optimisation technique linear programming is explained; it will be used to optimise connections between road parts.

3.1 Image Segmentation using Normalized Cuts

Image segmentation is a procedure which divides an image into non-overlapping regions of pixels having similar properties according to some criterion. It is often done as the first step for object extraction, because features derived from regions, for example the colour distribution and the shape, can be more meaningful and robust than features from single pixels.

Image segmentation is a difficult task for several reasons. It is an inverse problem; the information provided is not sufficient to solve the problem unambiguously. In addition, the "right" solution generally depends on the application. Sometimes a coarse segmentation is desired; sometimes the segmentation should be more detailed. Depending on the objects to be extracted in subsequent steps, regions might be perceived as belonging together or not. Therefore, a single segmentation method which could be used in every application does not exist. It is necessary to adapt the segmentation method to the application. Various segmentation methods have been developed, which differ by the degree to which they can be adapted to specific requirements of applications. For an early overview on basic segmentation strategies for different sources and objects, see (Eckstein, 1996). Lucchese and Mitra (2001) give a comprehensive overview on segmentation techniques with a focus on colour images and colour cues.

In many segmentation methods, only one criterion (for example the colour homogeneity) can be used. But for many applications, it is desirable to combine different criteria, in order to make the segmentation more significant for the application. It is also desirable to use global characteristics of the image in order to overcome small disturbances. These disturbances can cause disruptions in the regions when local segmentation approaches, such as region growing, are used. Another problem of local segmentation approaches is leaking between regions belonging to different objects where the contrast between them is low.

In this thesis, the normalized cuts method is used to perform an initial segmentation. This method has several advantages: it allows to integrate various features, and it aims at a globally optimal segmentation. These properties make it possible to incorporate extensive object knowledge into the segmentation. The Normalized Cuts method is a method for the division of undirected graphs into segments based on their weighted edges. The method and its application to image segmentation are described in (Shi and Malik, 2000).

The advantages of graph-based algorithms aiming at a globally optimal segmentation can be seen in a comparison of image segmentation algorithms by Estrada and Jepson (2009). The authors compare four current segmentation algorithms, evaluating the completeness and correctness of their boundaries over a variety of images and a range of the respective input parameters. The four algorithms are spectral embedding (SE) min-cut (Estrada and Jepson, 2004), normalized cuts, local variation (Felzenszwalb and Huttenlocher, 1998) and mean-shift (Comaniciu and Meer, 2002). Only SE min-cut and normalized cuts use global information. This makes the algorithms perform significantly better, especially in the presence of noise, as

the experiments done by Estrada and Jepson (2009) show. A disadvantage of both algorithms is the high computational complexity, which is reflected in a rather long computation time. In the comparison, the normalized cuts algorithm was significantly faster than the SE min-cut algorithm. The authors state that SE min-cut yields better results than normalized cuts in terms of boundary correctness – normalized cuts produced more spurious boundaries in their experiments. However, the SE min-cut algorithm as described by the authors contains a separate merging stage as the last step, while the other algorithms are run without a subsequent merging. It can be deduced that after a merging step, the border correctness of normalized cuts and SE min-cuts will probably be similar.

3.1.1 The Normalized Cuts Framework

In order to apply the normalized cuts method to image segmentation, the image is regarded as an undirected graph. The nodes of the graph are the pixels of the image; they are connected by weighted edges. The edge weights are measures for the similarity between pairs of connected pixels (Figure 3.1). Theoretically, each pixel is connected to all other pixels, but in practice only pixels lying close together have edge weights different from zero. There are two reasons for this: first, similarities for pixels lying far away from each other are not very significant for object extraction, and second, a fully connected graph is computationally intractable in practice. Figure 3.1 depicts a 4-neighbourhood of directly adjacent pixels, but usually a wider neighbourhood is used in order to reduce the influence of outliers.

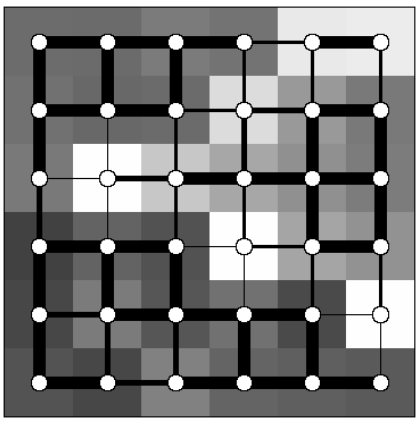


Figure 3.1. Image as graph. White disks are nodes, black lines are edges. For clarity, the connections displayed are only those in a 4-neighbourhood. The thickness of the edges symbolises the edge weight.

The similarity measure is defined according to the goals of the segmentation. The determination of a suitable similarity measure is a crucial part of the algorithm. It should allow an efficient separation of the target objects from the surrounding regions. For example, if the objects that are to be extracted have distinctly different intensities compared to their surroundings, intensity should be used to define the similarity measure. It is possible to combine several similarity criteria, which is often desirable when the separation cannot be realised using only one criterion. The weight as the combination of all similarity measures is usually normalised to the interval between 0 and 1.

One possibility to convert pixel similarity measures to weights is to use the measures as variables in an exponential function (Shi and Malik, 2000). For example, the conversion of an intensity difference d_I into a weight w would be computed as

$$w = e^{\frac{-d_I^2}{\sigma_I^2}}$$

The parameter σ_I controls the decrease of the weight function. Generally, if σ_I is large, the function decreases slowly, so pixel pairs with a higher dissimilarity (i.e. the value in the numerator) can still have a relatively high weight. The value of σ_I should be higher if the differences between pixel pairs vary more within single object regions. In (Shi and Malik, 2000), it is recommended to set this value to 10%-20% of the range of the feature differences, i.e. in the example the range of all intensity differences in the image.

The weights for the pixel pairs are inserted into a symmetric *weight matrix* (*W*) whose row and column dimensions both equal the number of pixels (Figure 3.2). In graph theoretic terms, this matrix is the adjacency matrix which represents the weighted graph, but it will be referred to as weight matrix in this text. An element w_{ij} of the matrix *W* represents the weight of the edge between the pixel numbered *i* and the pixel numbered *j*. The pixels of the image are numbered consecutively in row-major order.

node	1	2	3	4	5	6	7	8	9
1	1	0.98	0	0.88	0	0	0	0	0
2	0.98	1	0.67	0	0.93	0	0	0	0
3	0	0.67	1	0	0	0.67	0	0	0
4	0.88	0	0	1	0.80	0	0.82	0	0
W = 5	0	0.93	0	0.80	1	0.93	0	0.02	0
6	0	0	0.67	0	0.93	1	0	0	0.10
7	0	0	0	0.82	0	0	1	0.04	0
8	0	0	0	0	0.02	0	0.04	1	0.25
9	0	0	0	0	0	0.10	0	0.25	1

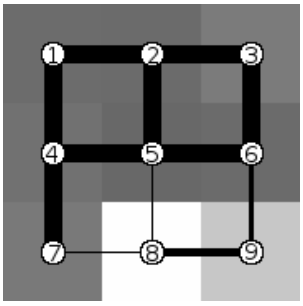


Figure 3.2. Weight matrix example for 3x3 pixel image. The thickness of the edges symbolises the edge weight. Pixel nodes are numbered consecutively. A 4-neighbourhood is used in this example.

For image segmentation, the graph representing the image is divided by removing some edges such that disconnected segments are formed. In order to separate dissimilar segments, the edges to be removed should have small weights. The first and simplest idea could thus be to minimise the sum of the weights of all removed edges, for example in order to partition the graph into two parts:

$$cut(A,B) = \sum_{i \in A, j \in B} w_{ij} \rightarrow \min \tag{3.1}$$

where *cut(A, B)* describes the sum of edge weights between two separate segments: node *i* lies in segment *A* and node *j* lies in segment *B*, w_{ij} is the weight of the edge between the nodes. The first application of this *minimum cut* criterion to image segmentation was presented by Wu and Leahy (1993). However, the minimum cut optimisation has a disadvantage: because of the absolute minimum criterion, it tends to keep the absolute number of separated edges small, which can lead to the separation of very small segments at the image borders. In order to overcome this disadvantage, Shi and Malik (2000) normalised the cut:

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(B,A)}{assoc(B,V)} \rightarrow \min \tag{3.2}$$

where *cut(A, B)* is defined as in Equation 3.1. *V* is the set of all nodes in the graph; *assoc(A,V)* is defined analogously to a *cut:*

$$assoc(A,V) = \sum_{i \in A, j \in V} w_{ij}$$

The difference between *cut* and *assoc* (short for association) is that the segments *A* and *V* are not disjoint, but *A* is a subset of *V*. The value of the denominator *assoc(A,V)* increases if the number of nodes in segment *A* increases. In this way, very small segments are avoided. Instead, the normalized-cut criterion favours segments that are compact and of approximately equal size. The normalized cuts criterion can be generalised to an arbitrary number *n* of segments:

$$Ncut(A_1,\ldots,A_n) = \sum_{i=1}^{n} \frac{cut(A_i, V \setminus A_i)}{assoc(A_i,V)} \rightarrow \min$$

The minimisation of the normalized cut does not only maximise the dissimilarity between different segments, it also maximises the similarity inside the segments at the same time, as shown in (Shi and Malik, 2000).

3.1.2 Calculation with Eigenvectors for a Partition into Two Segments

The minimum has to be found by an optimisation. Shi and Malik (2000) have shown that the minimisation of the normalized cut criterion (for the partition into two segments *A* and *B*) can be expressed as:

$$\min_{\vec{x}} Ncut(\vec{x}) = \min_{\vec{y}} \frac{\vec{y}^T (D-W)\vec{y}}{\vec{y}^T D \vec{y}} \tag{3.3}$$

$$\text{with constraints } y_i \in \{1, -b\} \text{ and } \vec{y}^T D \vec{1} = 0$$

$$\text{where } b = \frac{\sum_{i \in A} d_{ii}}{\sum_{i \in B} d_{ii}}$$

Here, the vector $\vec{x}$ is an indicator vector for the segmentation: its dimension corresponds to the number of nodes, and each element can take one of two discrete values (here 1 and -1). The values represent assignments to segments. *W* is the weight matrix, and *D* is the degree matrix, a diagonal matrix which contains the weighted degrees of the nodes, i.e. the diagonal entry d_{ii} is the sum of the weights in *W* belonging to the node *i*:

$$d_{ii} = \sum_j w_{ij}$$

The matrix $L = D - W$ is the *Laplacian matrix* of the graph. The vector $\vec{y}$ is an indicator vector corresponding to $\vec{x}$, whose elements may also take two discrete values; however, those values are 1 and -*b* rather than 1 and -1. The vector $\vec{1}$ is a vector where all elements have the value 1; its dimension equals the number of pixels. This minimisation problem for discrete

values is NP-hard (Shi and Malik, 2000), but if the first constraint, the condition for $\vec{y}$ to take only two discrete values, is relaxed, i.e. if the elements y_i may take real values, it can be solved approximately. The term

$$\frac{\vec{y}^T(D-W)\vec{y}}{\vec{y}^T D\vec{y}}$$

on the right hand side of Equation 3.3 is a generalised Rayleigh quotient (Shawe-Taylor and Cristianini, 2004), so its minimum with respect to $\vec{y}$ can be found by solving the generalised eigenvalue problem

$$(D-W)\vec{y} = \lambda D\vec{y} \tag{3.4}$$

for the smallest eigenvalue; the corresponding eigenvector minimises the Rayleigh quotient (Equation 3.3). In the following, the eigenvector corresponding to the smallest eigenvalue will also be called the smallest eigenvector, the eigenvector corresponding to the second smallest eigenvector will be called the second smallest eigenvector and so on. The generalised eigenvalue problem can be solved by transforming it into a corresponding symmetric standard eigenvalue problem (Shi and Malik, 2000):

$$D^{-1/2}(D-W)D^{-1/2}\vec{z} = \lambda\vec{z} \text{ with } \vec{z} = D^{1/2}\vec{y} \tag{3.5}$$

The smallest eigenvalue of this system is 0 because the Laplacian matrix is positive semi-definite, and so is the symmetric matrix $D^{-1/2}(D-W)D^{-1/2}$. The corresponding smallest eigenvector is $\vec{z}_0 = D^{1/2}\vec{1}$, so the corresponding eigenvector that solves the generalised eigenvalue problem is $\vec{y}_0 = \vec{1}$. However, this violates the second constraint of Equation 3.3 ($\vec{y}^T D\vec{1} = 0$), which means that the vector $\vec{y}_0$ does not constitute a feasible solution to the problem. A minimisation that satisfies the constraint can be found using the Courant-Fischer min-max theorem: if the Rayleigh quotient is to be minimised under the constraint that the vector that constitutes the solution is perpendicular to the smallest eigenvector, the solution is the second smallest eigenvector (Golub and Van Loan, 1996). The standardised system is minimised, yielding the second smallest eigenvector:

$$\vec{z}_1 = \underset{\vec{z}^T\vec{z}_0=0}{\arg\min} \frac{\vec{z}^T D^{-1/2}(D-W)D^{-1/2}\vec{z}}{\vec{z}^T\vec{z}}$$

from which the second smallest eigenvector of the generalised problem in Equation 3.4 is derived: $\vec{y}_1 = D^{-1/2}\vec{z}_1$.

The solution vector $\vec{y}_1$ can now be used to find the graph partitioning. It has to be discretised in order to obtain the desired indicator vector, which means one has to find a threshold for the eigenvector values. All nodes corresponding to eigenvector elements whose values are below the threshold are placed in one segment and the other nodes in the other segment:

$$\begin{aligned} n_i &\in A \text{ if } y_{1i} > t \\ n_i &\in B \text{ if } y_{1i} \le t \end{aligned} \tag{3.6}$$

where n_i denotes node i, y_{1i} denotes the value for element i in the second smallest eigenvector and t denotes the threshold. Shi and Malik (2000) mention several possibilities to choose the threshold. The authors recommend searching for a threshold that minimises the normalized cuts value of the partition: several possible thresholds, evenly spaced within the range of eigenvector values, are tested. For each of these thresholds the graph is partitioned (Equation 3.6) and the normalized cuts value of the partition (Equation 3.2) is calculated. The partition which yields the smallest normalized cuts value is chosen as the optimal partition. Another possibility is to set the threshold to zero.

As an example, consider the weight matrix from Figure 3.2. The diagonal of its degree matrix and its Laplacian matrix are

$$diag(D_{ii}) = \begin{bmatrix} 2.86 \\ 3.58 \\ 2.34 \\ 3.50 \\ 3.68 \\ 2.70 \\ 1.86 \\ 1.31 \\ 1.35 \end{bmatrix} \quad D - W = \begin{bmatrix} 1.86 & -0.98 & 0 & -0.88 & 0 & 0 & 0 & 0 & 0 \\ -0.98 & 2.58 & -0.67 & 0 & -0.93 & 0 & 0 & 0 & 0 \\ 0 & -0.67 & 1.34 & 0 & 0 & -0.67 & 0 & 0 & 0 \\ -0.88 & 0 & 0 & 2.50 & -0.80 & 0 & -0.82 & 0 & 0 \\ 0 & -0.93 & 0 & -0.80 & 2.68 & -0.93 & 0 & -0.02 & 0 \\ 0 & 0 & -0.67 & 0 & -0.93 & 1.70 & 0 & 0 & -0.10 \\ 0 & 0 & 0 & -0.82 & 0 & 0 & 0.86 & -0.04 & 0 \\ 0 & 0 & 0 & 0 & -0.02 & 0 & -0.04 & 0.31 & -0.25 \\ 0 & 0 & 0 & 0 & 0 & -0.10 & 0 & -0.25 & 0.35 \end{bmatrix}$$

The corresponding matrix for the symmetric standard eigenvector problem is

$$D^{-1/2}(D-W)D^{-1/2} = \begin{bmatrix} 0.65 & -0.31 & 0 & -0.28 & 0 & 0 & 0 & 0 & 0 \\ -0.31 & 0.72 & -0.23 & 0 & -0.26 & 0 & 0 & 0 & 0 \\ 0 & -0.23 & 0.57 & 0 & 0 & -0.27 & 0 & 0 & 0 \\ -0.28 & 0 & 0 & 0.71 & -0.22 & 0 & -0.32 & 0 & 0 \\ 0 & -0.26 & 0 & -0.22 & 0.73 & -0.30 & 0 & -0.01 & 0 \\ 0 & 0 & -0.27 & 0 & -0.30 & 0.63 & 0 & 0 & -0.05 \\ 0 & 0 & 0 & -0.32 & 0 & 0 & 0.46 & -0.03 & 0 \\ 0 & 0 & 0 & 0 & -0.01 & 0 & -0.03 & 0.24 & -0.19 \\ 0 & 0 & 0 & 0 & 0 & -0.05 & 0 & -0.19 & 0.26 \end{bmatrix}$$

For this matrix, the smallest eigenvalue is zero, the second smallest eigenvalue is 0.06, and the eigenvector corresponding to the second smallest eigenvalue is

$$\vec{z}_1{}^T = [-0.16 \quad -0.17 \quad -0.11 \quad -0.16 \quad -0.14 \quad -0.06 \quad -0.08 \quad 0.68 \quad 0.64]$$

According to Equation 3.4, this vector can be transformed back into the eigenvector of the generalised eigensystem:

$$\vec{y}_1{}^T = \left(D^{-1/2}\vec{z}_1\right)^T = [-0.10 \quad -0.09 \quad -0.07 \quad -0.08 \quad -0.07 \quad -0.04 \quad -0.06 \quad 0.59 \quad 0.55]$$

This vector clearly indicates the optimal partition; it can be easily discretised using either of the methods explained above, and the image is consequently partitioned into one segment that contains the first seven pixels and one segment that contains the last two pixels. Figure 3.2 shows this to be a reasonable partition for the image graph. In cases where the eigenvector values are more continuously distributed, the partition is more ambiguous; in such a case the best method to discretise the vector is the search for the threshold which minimises the

normalized cuts value. A very continuous distribution of values in the eigenvector should be treated as an indication that the image content does not warrant a partition.

3.1.3 Extension to More than Two Segments

For a segmentation into more than two segments, two general approaches are possible: a *recursive subdivision* of the already found segments or a division using more eigenvectors: for a partition into *k* parts, the *k* smallest eigenvectors are calculated. The latter method is called *k-way cut*. The recursive subdivision has to be stopped when an eigenvector has a continuous spectrum because a division based on such an unstable eigenvector would be quite arbitrary. Further subdivisions are not possible even if they would be desirable. The *k*-way cut, on the other hand, uses all the *k* smallest eigenvectors simultaneously. Thus, even if one or more eigenvectors are unstable, all eigenvectors will be used, which might lead to a finer partition (Shi and Malik, 2000). In total, both methods are suitable, but the *k*-way cut is computationally more economical because several ordered eigenvectors of one matrix can be computed faster than one eigenvector of several matrices.

For the k-way cut, the minimisation criterion is

$$\min_{Y} \sum_{i=1}^{k} \frac{Y_i^T (D-W) Y_i}{Y_i^T D Y_i} = \min_{Y} \frac{tr\left(Y^T (D-W) Y\right)}{tr\left(Y^T D Y\right)} \tag{3.7}$$

where $Y = [Y_1 \quad \ldots \quad Y_k]$ contains the k smallest eigenvectors. The expression *tr(.)* denotes the trace of a matrix. When using the *k*-way cut, the set of *k* smallest eigenvectors has to be discretised to find the optimal partition. This discretisation is not as trivial as in the segmentation into two parts where only one eigenvector has to be discretised. An optimal discretisation has to be found for the set of all *k* eigenvectors together in order to prevent inconsistencies (each pixel has to be assigned to one and only one segment). A method to find a discrete optimum is described in (Yu and Shi, 2003). The authors note that the eigenvector solution, while being an optimal solution to a minimisation of the generalized Rayleigh quotient, is not unique. Any rotation of the eigenvectors constitutes an optimal solution, which can be seen by substituting a set of rotated vectors *YR* for *Y* in the term to be minimised in Equation 3.7:

$$\frac{tr\left((YR)^T (D-W) YR\right)}{tr\left((YR)^T D YR\right)} = \frac{tr\left(R^T Y^T (D-W) YR\right)}{tr\left(R^T Y^T D YR\right)} = \frac{tr\left(RR^T Y^T (D-W) Y\right)}{tr\left(RR^T Y^T D Y\right)} = \frac{tr\left(Y^T (D-W) Y\right)}{tr\left(Y^T D Y\right)}$$

The equation is valid because the trace of a matrix product is invariant under cyclic permutations and for rotation matrices $RR^T = I$ holds true. It follows that the normalized cuts value is the same for any rotation of the eigenvectors *Y*. Therefore, Yu and Shi (2003) propose to find a discrete solution that lies closest to a rotated non-discrete optimal solution:

$$\begin{aligned} &\text{minimise} && \left\| X - YR \right\|^2 \\ &\text{subject to} && X_{ij} \in \{0,1\} \; \forall i = 1,\ldots,n; j = 1,\ldots,k \\ & && X \cdot \vec{1} = \vec{1} \\ & && R^T R = I \end{aligned}$$

Here, $\|A\| = \sqrt{tr(AA^T)}$ denotes the Frobenius norm of matrix A. $Y = [\vec{y}_1 \ldots \vec{y}_k]$ contains the k eigenvectors which are normalized such each row of Y, which represents a vector in k-space, is normalized to a length of 1. $X = [\vec{x}_1 \ldots \vec{x}_k]$ contains the k discrete indicator vectors that are to be determined, with the condition that for each node i = 1,...,n one and only one corresponding indicator element is set to 1, the others are 0. R is the unknown rotation matrix for the eigenvectors which moves them closest to the discretised vectors. The optimum is found iteratively by alternately optimising R and X. In the optimisation step for X the rotated vectors $Y' = YR$ are discretised by nonmaximum suppression over the rows of Y':

$$X_{ij} = \begin{cases} 1 & \text{if } j = \arg\max_k Y'_{ik} \\ 0 & \text{else} \end{cases}$$

In the optimisation step for R the rotation matrix must be found which comes closest to mapping Y on X. This is called an *orthogonal Procrustes problem* and can be solved by singular value decomposition of the matrix product $X^T Y$ (Golub and Van Loan, 1996):

$$X^T Y = USV^T \quad \text{(singular value decomposition)}$$
$$R = VU^T$$

The matrices U, S and V are the factors of the singular value decomposition. U and V are orthogonal matrices, S is a diagonal matrix whose entries are the singular values of $X^T Y$. The result is a set of discretised vectors, as many as the desired number of segments, which has to be specified beforehand. Each vector indicates one segment by its nonzero values.

One problem of the normalized cuts method is that the desired number k of segments must be given to the algorithm as input. This means that the desired number of segments must in effect be known before the segmentation, which is usually not the case. For general clustering purposes, the optimal number of clusters can often be determined by finding the maximal *eigengap* – the difference of the k^{th} and $(k+1)^{th}$ eigenvalue (Azran and Ghahramani, 2006). For real-world images, however, this criterion is usually not applicable because no unequivocal optimal segmentation exists and consequently, the eigengaps might not differ much from each other. Therefore, the number of segments is often chosen empirically. It is advisable to choose a relatively high number of segments, yielding an oversegmentation, and group the segments afterwards. In this way, the number of segments does not need to be adjusted for each new image. An oversegmentation with subsequent grouping is also useful to deal with the tendency of the normalized cuts algorithm to yield equally-sized, compact segments.

The calculation of the eigenvectors is computationally expensive, but since the Laplacian matrix is sparse and symmetric, the *Lanczos method* (Golub and Van Loan, 1996) can be used for the eigenvector calculation. The Lanczos method is an iterative method with which a small set of eigenvectors belonging to the largest or smallest eigenvalues for a symmetric matrix can be effectively computed. The Lanczos algorithm calculates a symmetric tridiagonal matrix T from a symmetric matrix A such that $T_{m\times m} = Q^T_{n\times m} A_{n\times n} Q_{n\times m}$, where $Q = [\vec{q}_1 \ldots \vec{q}_m]$ is orthogonal. In each iteration step i, one new vector $\vec{q}_i$ and two new coefficients of the tridiagonal matrix T are calculated, such that the dimensions m of T grow with each iteration step. The eigenvalues of T approximate some of the eigenvalues of A, and the first eigenvalues to converge in this way are the largest and the smallest eigenvalues of A.

Therefore, the required dimension of T is much smaller than the dimension of A if only some of the extremal (smallest or largest) eigenvalues are needed. The eigenvalues and eigenvectors of T are calculated by spectral decomposition, which is very efficient for tridiagonal matrices. The spectral decomposition yields $T = BCB^T$, where B is an orthogonal matrix containing the eigenvectors of T, and C is a diagonal matrix containing its eigenvalues. A set of extremal (largest and smallest) eigenvectors of A can now be calculated: $V_{n\times m} = Q_{n\times m} B_{m\times m}$, where $V = [\vec{v}_1 \ldots \vec{v}_m]$ contains the eigenvectors.

One advantage of the normalized cuts method is the possibility to combine several different features in one step by incorporating them into the similarity weights. This is a property that is important in complex surroundings. The difficulties that arise from the task of combining the results of several segmentation steps can be avoided in this way. Another advantage is that the algorithm takes both local and global characteristics into account. Local characteristics are incorporated in the weight matrix which contains the weights of neighbouring pixels. In this way the similarity of pixels in a close neighbourhood is accounted for. Global characteristics are considered when the optimal cut is calculated: a global minimum criterion must be met. This is a considerable advantage of the method, because in this way, small disturbances such as short or weak edges are ignored by the algorithm.

3.2 Linear Programming

Linear programming is a technique for the optimisation of a specific type of problem. Optimisation in general is concerned with finding the parameters of a system, called *variables* or *unknowns*, which define the best solution to a problem in terms of maximising or minimising an *objective function* (Nocedal and Wright, 2006). The optimisation is *constrained* if the variables have to satisfy certain conditions, usually formulated as equations or inequalities. If the variables may take integer values only, the optimisation is *discrete*. Discrete optimisation problems are usually more difficult to solve than continuous optimisation problems. The general mathematical formulation for an optimisation problem is as follows:

$$\min f(\vec{x}) \quad \text{subject to} \quad \begin{array}{l} c_i(\vec{x}) = 0,\ i \in E \\ c_i(\vec{x}) \geq 0,\ i \in I \end{array} \tag{3.8}$$

where f denotes the objective function, $\vec{x}$ is a vector containing the variables, and c_i are the constraint functions, which belong either to the set of equality constraints E or the set of inequality constraints I. The optimisation can be reformulated in order to maximise the objective function. If both the set E and the set I are empty the optimisation is unconstrained. For more information about optimisation in general, refer to Nocedal and Wright (2006), who give a comprehensive introduction into several optimisation methods.

Linear programming was developed in the 1940s. It was one of the first important applications of the then new computer technology and continues to be the most widespread optimisation method, having applications in many fields from economics to engineering. Linear programming can be applied for a *linear* objective function which is subject to constraints that can be expressed with *linear* equations or inequalities (Dantzig and Thapa, 1997). Being linear, the general formulation of the optimisation can be expressed in matrix form; it is usually stated in the *standard form* (Nocedal and Wright, 2006), also called *slack form* or *augmented form*:

$$\min \vec{w}^T \vec{x} \quad \text{subject to} \quad A\vec{x} = \vec{b}; \quad x_i \geq 0 \; \forall \; i = 1,...,n \tag{3.9}$$

As in Equation 3.8, the vector $\vec{x}$ contains the variables. The vector $\vec{w}$ contains the coefficients of the linear objective function. The *constraint matrix A* and the vector $\vec{b}$ contain the coefficients of the linear constraint functions; n is the number of variables. Any formulation of a linear program can be converted into the standard form: maximising an objective function is equivalent to minimising its negative. Inequality constraints can be converted into equality constraints by adding so-called *slack variables*: for each inequality constraint a new variable is introduced, for example if constraint k is a greater-than inequality, the slack variable s_k is subtracted:

$$a_{k1}x_1 + ... + a_{kn}x_n \geq b_k \rightarrow a_{k1}x_1 + ... + a_{kn}x_n - s_k = b_k$$

In subsequent calculations, the slack variables are added to the vector $\vec{x}$ and treated in the same way as the other variables. Another need for conversion occurs if a variable is unrestricted (allowed to take negative values). As the linear programming formulation only allows positive values for variables, an unrestricted variable must be replaced by two new variables for which the nonnegativity condition holds. The difference of these variables expresses the original unrestricted variable, for example if x_1 should be unrestricted, it is replaced by the variables y_{11} and y_{12}:

$$a_{k1}x_1 + ... + a_{kn}x_n = b_k \rightarrow a_{k1}(y_{11} - y_{12}) + ... + a_{kn}x_n - s_k = b_k$$

If a local optimum for a linear programming problem has been found, it is also the global optimum.

A simple linear program with only two variables will be used to demonstrate some of the properties of linear programming problems and strategies for their solution:

Suppose that you have to travel along a river for 200 km, and you have 8 hours time to reach your destination. You can take a bus or a boat for any part of the trip, you can change the means of transport at any place and it is assumed that the time loss involved in changing the means of transport is negligible. The bus fare per km is twice as much as the boat fare; the bus moves with an average speed of 60 km/h, and the boat with an average speed of 10 km/h. You want to minimise your travelling expenses.

This problem can be formulated as a linear program as follows:

$$\begin{aligned} &\text{objective function:} && 2x_{bus} + x_{boat} \rightarrow \min \\ &\text{subject to} && c_1: \quad x_{bus} + \quad x_{boat} = 200 \\ & && c_2: \frac{1}{60}x_{bus} + \frac{1}{10}x_{boat} \leq 8 \\ &\text{and} && x_{bus} \geq 0, x_{boat} \geq 0 \end{aligned} \tag{3.10}$$

where x_{Bus} and x_{Boat} are the unknown variables: the distances (in km) travelled by bus and by boat, respectively. A two-dimensional linear program like this can be displayed and solved graphically. Figure 3.3 shows how the example program is set up and solved graphically.

Constraint 1, the equality constraint, is depicted as a line, and constraint 2, the inequality constraint is depicted as a region. The intersection of both constraints defines the feasible region (red line); any solution must lie inside the feasible set in order to satisfy all constraints. Some contour lines of the objective function are shown, depicted as dashed lines. The arrow indicates the direction in which the value of the objective function decreases. The solution to this linear program can be found as the point of the feasible region where the objective function has the lowest value (red circle).

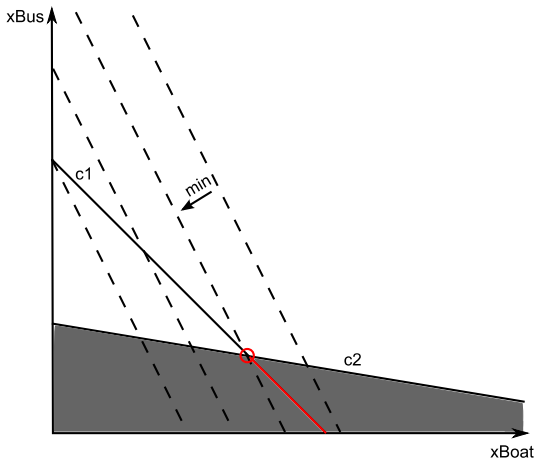


Figure 3.3. Graphical solution of linear program. Line c_1: constraint 1. Region c_2: constraint 2. Red line: feasible region. Dashed lines: contour lines of objective function. Red circle: optimal solution.

In linear problems where all constraints are inequalities, the feasible region consists not only of a line but of a region bordered by straight lines and vertices (Figure 3.4). From the diagram it is clear that the optimal solution must lie at a vertex of the polygon which delimits the feasible region. If the slope of the objective function is the same as the slope of a constraint function at which an optimal solution is found, there will be no unique optimal solution: every point along one edge constitutes an optimal solution. Obviously, there will be no solution if there is no feasible region or if the feasible region is unbounded towards the minimum.

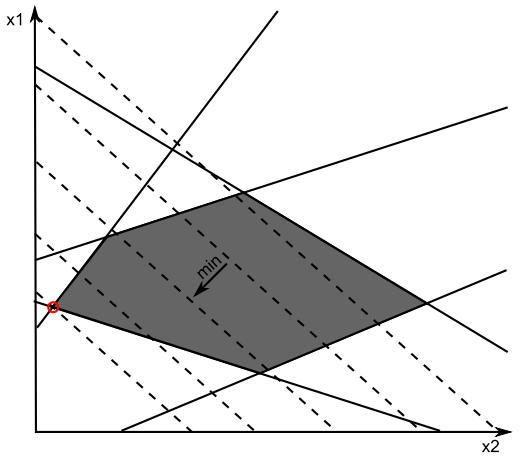


Figure 3.4. Feasible region (grey) of a two-dimensional linear program with five inequality constraints. Continuous lines: borders of half-planes for constraints. Dashed lines: contour lines of objective function. Red circle: optimal solution.

In any number of dimensions, the feasible region (if it exists and is bounded) is always a polytope[1], and the optimal solution lies at a vertex of this polytope. If the optimal solution is not unique, any weighted average of vertices that constitute optimal solutions will also be optimal solutions (for a proof see Dantzig and Thapa, 1997). This is equivalent to saying that if the optimal solution is not unique, all optimal solutions lie on an edge, a face or a facet between the vertices that are optimal solutions. Even the whole feasible region can consist of optimal solutions. The vertices of the feasible region are also called *basic feasible points* or *basic feasible solutions*.

3.2.1 The Simplex Method

The algorithm commonly used to solve linear programming problems is the simplex method, which uses the concept of the basic feasible points. It was developed primarily by G.B. Dantzig (first published in Dantzig, 1951). The basic principle of the simplex method is to move along the boundaries of the feasible region from vertex to vertex until the optimal vertex is found. The simplex method is illustrated by solving the example in Equation 3.10.

The linear program must be in the standard form (Equation 3.9); if necessary, it must be converted into the standard form. In the example, the inequality constraint is converted into an equality constraint by adding a slack variable to obtain the following set of linear equations:

$$\begin{aligned} 2x_{bus} + \quad x_{boat} \quad &= z \\ x_{bus} + \quad x_{boat} \quad &= 200 \\ \frac{1}{60}x_{bus} + \frac{1}{10}x_{boat} + s &= 8 \end{aligned}$$

$$\text{with the nonnegativity constraints } x_{bus} \geq 0,\ x_{boat} \geq 0, s \geq 0$$

The system of equations has four variables: x_{bus} and x_{boat} as defined above, the additional slack variable s and the value of the objective function z.

As the simplex method moves from one basic feasible point to the next, it requires a basic feasible point as starting point. To find one basic feasible point and start the simplex method, the system is converted into a *canonical form*. In the canonical form, each equation contains one variable, called *basic variable*, that does not occur in any other equation. The coefficients of the basic variables are 1. All other variables can occur in more than one equation with any coefficient and are called *non-basic variables*. In general, the canonical form can be written as

$$\begin{aligned} -z + \vec{0}^T \vec{x}_B + \vec{w}^T \vec{x}_N &= -z_0 \\ I\vec{x}_B + \quad A\vec{x}_N &= \vec{b} \end{aligned} \quad \text{with the basic feasible solution} \quad \begin{aligned} \vec{x}_B &= \vec{b} \geq 0; \quad \vec{x}_N = 0 \\ z &= z_0 \end{aligned} \quad . \quad (3.11)$$

The first row shows the objective function converted to canonical form. The unknowns are divided into a vector of basic variables $\vec{x}_B$ and a vector of non-basic variables $\vec{x}_N$. z is the variable for the value of the objective function. The vector $\vec{w}$ contains the current coefficients of the non-basic variables in the objective function, also called *relative cost factors*. The coefficients of the basic variables are always 0 in the current objective function. On the right hand side, the value z_0 gives the current value of the objective function. The second row shows the set of linear equations formed from the constraints, again divided into a basic and a

[1] A polytope is a generalisation of a polygon: an n-dimensional object with flat sides.

non-basic part. *I* denotes the identity matrix: all basic variables have unit coefficients. *A* is a matrix that contains the current coefficients of the non-basic variables. The vector $\vec{b}$ contains the current values of the right hand side, which are at the same time the values for the basic variables in the current solution.

In the example, the canonical form can be found simply by eliminating the variable x_{bus} from the first and the third equation:

$$\begin{aligned} -z \qquad\qquad - x_{boat} &= -400 \\ x_{bus} + \qquad\quad x_{boat} &= 200 \\ s + \frac{1}{12} x_{boat} &= 4\frac{2}{3} \end{aligned} \tag{3.12}$$

The canonical form in Equation 3.12 has three basic variables: z, x_{bus} and s, and one non-basic variable: x_{boat}, which can be found in more than one equation. The non-basic variable is set to 0, which yields the following values for the basic variables: $z = 400$, $x_{bus} = 200$, $s = 4\ 2/3$. The values for the variables are nonnegative, which means that this solution is feasible. With this solution, the simplex algorithm is started. If the solution is not feasible, another start solution must be found. For larger linear programs, the canonical form and the feasible solution are usually found by solving an auxiliary linear program which is easy to solve and yields a basic feasible solution, if such a solution exists at all.

Now, the steps of the simplex method are performed as follows:

1. The current solution is first *tested for optimality*. This can be determined by examining the relative cost factors: the coefficients of the objective function in the current canonical form. The variables in the modified objective function are all non-basic (apart from *z*), which means their values are 0 at this point. The solution is optimal if all relative cost factors are nonnegative: $w_j \geq 0 \quad \forall\ j = 1...m$ with *m* as the number of non-basic variables. But if any of the coefficients are negative, the solution is not optimal because a lower value for *z* can be achieved by increasing the value of a variable with a negative coefficient. If the solution is optimal, the algorithm stops, and the current basic feasible solution is the optimal solution. If the lowest relative cost factor is 0, the optimal solution is not unique: the value of the objective function cannot decrease further, but the set of basic feasible points can be changed without changing the value of the objective function. If the solution is not optimal, the algorithm continues with the next step. In the example, the coefficient of the non-basic variable x_{boat} is negative, so the solution can be improved by increasing the value of x_{boat}.
2. The algorithm now attempts to improve the solution by changing the set of basic variables. First, the *new basic variable* (*incoming* or *entering variable*) is selected from the set of non-basic variables: setting one basic variable to 0, which becomes non-basic, and increasing the value of one non-basic variable, which becomes basic. Several strategies are possible to decide which variable to select. A simple strategy is to choose the variable with the lowest relative cost factor: the index of the new variable is

$$s = \arg\min_j w_j$$

   In the example, the only non-basic variable x_{boat} will become the new basic variable.

3. It is checked whether the solution with the new basic variable is *unbounded*. The solution is unbounded if the value of the new basic variable x_{Ns} can be increased to infinity while the solution stays feasible (i.e. all other variables stay ≥ 0). This is the case if all coefficients of x_{Ns} are ≤ 0: $A_{js} \le 0 \quad \forall\, j = 1...m$. If the solution is unbounded, the algorithm stops: in this case there is no finite optimal solution. In the example, two coefficients of x_{boat} are positive (in the second and third equation of Equation 3.12), so the new solution will be bounded and the algorithm can proceed.
4. Next, the *new non-basic variable* (*outgoing* or *leaving variable*) and the value of the new basic variable are determined. The value of the new basic variable is

$$x_s = \min_{\{j \mid A_{js} > 0\}} \frac{b_j}{A_{js}} = \frac{b_r}{A_{rs}} \ge 0$$

where *r* is the row index of the outgoing variable. In the example, we have (following from Equation 3.11):

$$x_{boat} = \min\left(\frac{200}{1}, \frac{4\,2/3}{1/12} \right) = \min(200, 56) = 56\,.$$

The minimum occurs in the last one of the constraint equations, so the outgoing variable is taken from that equation: the new non-basic variable is *s*. If there is more than one index *j* yielding the minimum value, the choice of the outgoing variable among them is arbitrary. But if the minimum value is 0 for more than one index *j*, which is called the *degenerate case*, special precautions have to be taken to avoid *cycling* (which means that the algorithm is caught in a non-terminating loop). One possible strategy is Bland's rule:

$$\text{index of incoming variable: } s = \min_{\{j \mid w_j < 0\}} j$$

$$\text{index of outgoing variable: } r = \min_{\{j \mid x_s = \min\}} j$$

5. The linear program is transformed into a new canonical form with the new sets of basic and non-basic variables, and the procedure is repeated, starting with the *test for optimality*, until the optimal solution is found. In the example, the new canonical form is

$$\begin{aligned} -z \qquad & +12s = -344 \\ x_{bus} \quad & -12s = 144 \\ x_{boat} \quad & +12s = 56 \end{aligned}$$

The only relative cost factor in the current objective function is the coefficient of *s*. The value of the coefficient is +12, which means that the solution is optimal because the relative cost factor is positive: the value of the objective function cannot be further diminished by increasing the value of *s*. The algorithm terminates, and the solution to the linear program is: travel x_{bus} = 144 km by bus and x_{boat} = 56 km by boat.

The simplex method is very efficient for many practical problems that can be formulated as linear programs. It is possible that the number of steps required grows exponentially with

higher dimensions, but Dantzig and Thapa (1997) state that this rarely occurs in real-world problems. However, alternatives to the simplex method exist that avoid the exponential growth problem, for example interior point methods, which move through the feasible set rather than along the edges to find the optimal solution.

The *revised simplex method* (Dantzig and Thapa, 1997) is the method usually applied in general-purpose linear programming solvers. The strategy of the algorithm is the same but the computation exploits the fact that for many large linear programs the constraint matrix is sparse. Thus, the time and memory requirements can be reduced.

In many applications the variable values are restricted to be integers. However, the simplex method generally does not consider such restrictions. Some modified algorithms were developed to deal specifically with such restrictions (integer programming; cf. (Nocedal and Wright, 2006)), which are more computationally expensive. But there are some types of linear programs, characterised by specific properties of the constraint matrix, that guarantee a solution consisting of integer values for all variables. One important case are *unimodular matrices*. An integer program can be solved as a linear program if the constraint matrix, which contains the coefficients of the constraints, is totally unimodular and the right hand side (the vector $\vec{b}$ in Equation 3.9) consists only of integers (Papadimitriou and Steiglitz, 1998). A matrix A is totally unimodular if

1. every column of A has at most two non-zero entries,
2. every entry of A is either 0, 1 or -1,
3. it can be partitioned into two disjoint sets of rows B and C such that if two non-zero entries in a column have the same sign, the row of one is in B and the row of the other is in C, and
4. using the same partition into B and C, if two non-zero entries in a column have opposite signs, both rows are either in B or in C.

If the constraint matrix has these properties and the right hand side consists of integers, all basic feasible solutions will have only integer values for the variables.

4 A New Road Extraction Approach

4.1 Overview

From the literature overview (Chapter 2), it can be deduced that there is not one perfect road extraction method which is valid for all scenes and data sets. The method has to be adapted to the scene and the data because roads have different appearances in different types of scenes. Traditionally, three different types of scenes, called *global context* (Baumgartner et al., 1999), are distinguished in the scope of road extraction: urban areas, rural areas and forest areas. In this thesis, a part of the urban context area is considered: *suburban areas*.

Suburban areas (Figure 4.1) are characterised by low building density; they are often located in the outskirts of a city. The majority of buildings are detached houses or semi-detached houses which usually neither stand very close to each other nor very close to the roads; vegetated areas around the houses are frequent. The *roads* themselves have widths typically differing between 5 and 8 m, but in aerial images they often appear to be wider because of pavements and cycle lanes which often do not differ from the roads in colour. Depending on the widths of the pavements this can increase the perceived road width to approximately 15 m. Roads in suburban areas typically do not have road markings, apart from some main roads. Roads in suburban areas are almost always at the same level as the ground; bridges and embankments are rare. The appearance of the roads can be influenced by context objects in their vicinity. These are primarily vehicles, trees and buildings, and the shadows they may cast. Traffic is relatively sparse in suburban areas, so there are usually not many vehicles on the roads, sometimes none. Parking vehicles on the sides of roads may occur more often. Trees can be found on the sides of roads. They may occlude roads partially or even totally. Avenues with trees on the central reservation are extremely rare in suburban areas. Buildings in suburban areas are usually not high enough to directly occlude roads in aerial images, but their shadows can fall on the roads, as well as the shadows of trees, giving an abrupt and large difference in surface appearance. Buildings can also, at least locally, have similar radiometric properties to those of roads.

Figure 4.1 Examples of suburban scenes.

Roads are needed to provide fast connections between different places as well as access to all buildings and places of interest in one area. In suburban areas, the latter function of roads is prevalent. As a consequence, the roads in suburban areas are not designed for heavy traffic; they are rather narrow and short. Dead ends can be frequent, and the road network does not necessarily favour short connections between places.

Following the conclusions from the literature review, a *region-based road model* is used together with high resolution (approx. 0.1 m GSD) CIR orthoimages and, if available,

corresponding DSMs. In the following text, a *road* is defined as a part of the road network between two junctions, or between one junction and a dead end. Roads are modelled as elongated regions of approximately constant width that corresponds to the range of typical road widths for suburban areas. The surface is assumed to be relatively homogeneous but no specific colour is required. Because of the complexity of the scene and the expected disruptions by context objects, the roads are not required to be extracted in one piece from junction to junction: there can be gaps between extracted *road parts* in the first stage of the extraction. In later stages, the road parts are connected to *road subgraphs* (Figure 4.2) and finally to a road network.

Context objects that are considered in this work are vehicles, trees, buildings, vegetated areas and road surface areas. Vehicles can be found on roads or next to roads. Trees can be found next to roads; they can also occlude the road by overshadowing it. Buildings and vegetated areas, on the other hand, can never be found on roads, only next to them. Road surface areas can also occur on parking lots besides the roads. In the context of the road extraction algorithm presented here, road surface areas are considered as context objects in gaps between extracted road parts.

DSMs are used in this approach because they can provide very useful information for the extraction of roads as the roads are usually on the ground. Therefore, DSMs provide a valuable means to prevent false extractions, especially of building roofs. However, as DSMs are not always available, the road extraction can also be done without a digital surface model, at the expense of more false extractions. The resolution of the DSM does not need to be as high as for the image; a grid resolution of 0.5 m is sufficient, and the vertical resolution should be high enough to distinguish between the levels “ground” and “objects above ground” such as buildings and trees. This means that a vertical resolution of 1 m is sufficient. The DSM (containing high vegetation and buildings along with the terrain relief) will be converted into a normalised DSM (nDSM). A digital terrain model (DTM) is derived from the DSM by a morphological opening operation. The DTM is then subtracted from the DSM, yielding the nDSM.

The fact that the *road network function* to provide fast connections is limited in suburban areas is considered for the generation of the road network: connections are searched for locally, junction by junction, rather than globally by searching for the fastest connections.

The road extraction process consists of the following steps, described in detail in the subsequent sections:

1. Following the region-based road model, the process starts with a *segmentation* of the image (Section 4.2). The normalized cuts method (Section 3.1) is used for segmentation; the segmentation criteria are radiometric in nature, and knowledge about the appearance of roads is already used in the segmentation as far as possible. The aim is to have as many segment borders to coincide with the road borders as possible.
2. After the segmentation, a *grouping* of the segments follows (Section 4.3), using radiometric and geometric criteria, including a height criterion if a DSM is available, to compensate for the oversegmentation. The combination of the grouping criteria is based on fuzzy control (Zadeh, 1965; Klir, 2006).
3. From the grouped segments, *road part hypotheses* are extracted (Section 4.4) by classifying the segments as road segments or non-road segments, again using radiometric and geometric criteria, including a height criterion, if available. A road part can cover a road from junction to junction but is not required to.

4. Road part hypotheses (in the following called *road parts* for brevity) are connected to *road subgraphs* (see Figure 4.2) by *connecting lines* in the next step (Section 4.5) if they are likely to belong to the same road. The term subgraph suggests that the road subgraphs do not represent a global road network but rather a local part of the network. *Branches* can occur in the road subgraph, especially if false extractions are present.
5. Therefore, the subgraphs are optimised in the next step (Section 4.6) with the goal to remove the branches and obtain single *road strings*. Context objects are used to evaluate the branches for the optimisation. The optimisation is carried out using linear programming (Section 3.2).
6. In the last step, the *road network* is generated (Section 4.7). This comprises the search for junctions and the elimination of false hypotheses.

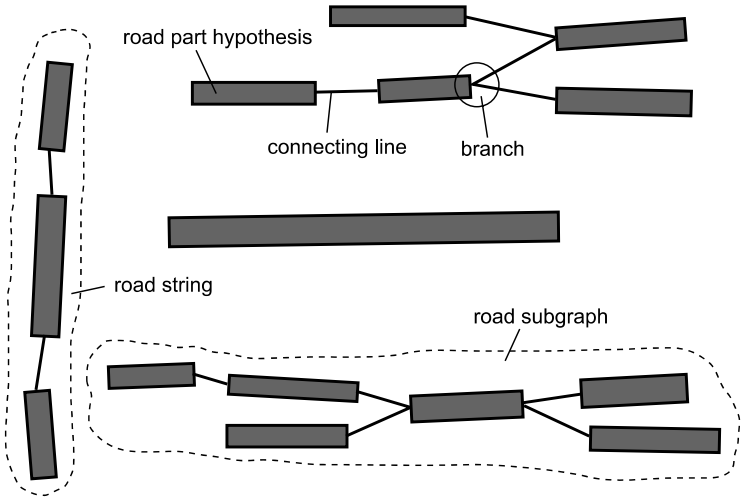


Figure 4.2. Schematic diagram of road parts, road subgraphs and road strings.

4.2 Segmentation using Normalized Cuts

The first step in the road extraction algorithm is the initial segmentation of the image. The goal of the initial segmentation is to divide the image into segments in such a way that all road borders correspond to segment borders: ideally, one segment should only contain either road pixels or non-road pixels, but not both. The suburban scenes from which roads are to be extracted are quite complex, and the roads do not have one single characteristic that sets them apart from their surroundings. Therefore, several criteria should be used which are derived from knowledge about the appearance of roads (the road model). Global characteristics of the image should also be used in order to prevent small disturbances (such as small areas with differing intensity) from distorting the segmentation. The normalized cuts algorithm (cf. Chapter 3.1) is used for the segmentation because it allows to use several criteria and aims at a global optimisation of the segmentation.

4.2.1 Weight Matrix

The normalized cuts algorithm uses similarities between pairs of pixels to determine the optimal segmentation. Similarities between pixel pairs are not considered across the whole image but only within a predefined neighbourhood around each pixel. The size of the neighbourhood is not limited to the direct neighbours of a pixel. In the presence of

disturbances of the surface, larger neighbourhoods reflect the local relations of a pixel better than only direct neighbourhoods: they have a smoothing effect. In this thesis, the neighbour pixels of a pixel are those lying inside a square with an edge length of $2n+1$ pixels (n pixels to the left, right, up and down of the centre pixel). For practical reasons (to keep the computational complexity manageable) the similarities to the centre pixel are only computed for a randomly selected subset of p_n% of the pixels in the neighbourhood of the centre pixel. The pixel pairs for which similarities are computed will be called *active pixel pairs*. For all other pixels, the similarity weight is set to zero.

The weight matrix W (cf. Chapter 3.1) is needed for the segmentation with normalized cuts. It contains the similarities between all active pixel pairs. The selection of similarity measures that are combined to generate the weight of each edge in the graph is based on the road model.

The similarity criteria used here are:

- Presence and strength of edges between two pixels
- Colour difference
- Hue difference
- NDVI difference

As roads are separated from their surroundings by edges, the first similarity criterion is the existence and strength of *edges between two pixels*. If there is an edge between two pixels, the pixels are considered to be dissimilar. The value of the similarity measure depends on the edge strength: higher edge strength leads to a lower similarity.

The edge criterion is calculated in two steps: in the first step it is determined whether an edge exists between two pixels; in the second step the edge strength is determined if an edge exists. To determine whether an edge exists between two pixels, the Laplacian of Gaussian (LoG) operator (Marr and Hildreth, 1980) is applied to the intensity image with the standard deviation σ_{LoG}. The LoG operator yields an image of the second derivatives of the intensity, where edges are indicated by sign changes (zero-crossings). The image of the second derivatives is converted into a binary image such that it only contains the sign information. For each active pixel pair it is checked whether a sign change occurs along the connecting line between the two pixels.

If a sign change is detected between two pixels, the edge strength at this point is taken from an edge amplitude image containing the absolute values of the image gradients after a Gaussian smoothing with the standard deviation $\sigma_G = \sigma_{LoG}$. Since the neighbourhood can comprise more than the directly adjacent pixels, the connecting line between the two pixels can lead through several other pixels. Only the two pixels that lie side by side on both sides of the sign change line are used to determine the edge strength value; it is the average edge amplitude of these two pixels. Figure 4.3 illustrates this combination of sign change and edge strength.

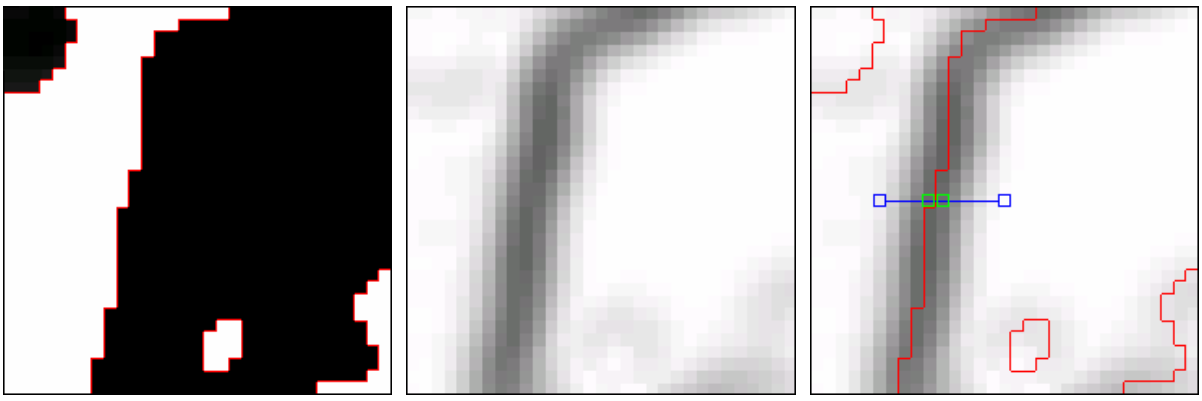


Figure 4.3. Combination of LoG image and edge amplitude image. Left: Binary LoG image. Centre: Edge amplitude image; low grey values denote high edge strength. Right: Edge amplitude image with lines of sign change from LoG in red. Pixels with blue frames: neighbouring pixels connected by blue line. Pixels with green frames: pixels from which the edge strength is calculated.

The combination of edge strength and location of sign change as opposed to the edge strength alone has two advantages. One is that pixels are only considered dissimilar with regard to the edge image if the pixels lie on different sides of the edge. That way, the position of the edge is more clearly incorporated into the weight matrix. The other is that no threshold for the edge strength is needed to determine if an edge is present.

If an edge is found, the average edge strength is converted to the edge weight using the following weight function, to obtain the first component w_{edge} of the weight between two pixels i and j:

$$w_{edge}(i,j) = \begin{cases} e^{\frac{-es(i,j)^2}{\sigma_{es}^2}} & \text{if sign change between pixels} \\ 1 & \text{otherwise} \end{cases} \qquad (4.1)$$

Here, $es(i, j)$ is the measured edge strength between the pixels i and j, and σ_{es} is a parameter that controls the decrease of the edge weight function. σ_{es} is set to 10% of the range of the edge intensity, as recommended in (Shi and Malik, 2000).

The second criterion is the *colour similarity* because roads usually have homogeneous surfaces, which means that the colour values of two pixels on a road surface should be similar. A measure for the colour similarity of two pixels is the length of the distance vector of their associated vectors in colour space. A shorter distance vector indicates a more similar colour. Many different colour spaces (e.g. Burger and Burge, 2006) can be derived from the original colour channels, but here, the differences between results with different colour spaces are not significant. Therefore, the original CIR colour space is used to calculate the colour similarity; it also gave slightly better results than other colour spaces (e.g. the HSI space). The weight for the colour similarity is then calculated in a manner analogous to Equation 4.1:

$$w_{colour}(i,j) = e^{\frac{-d_c(i,j)^2}{\sigma_c^2}}$$

Here, $d_c(i,j) = \left\| \vec{c}_i - \vec{c}_j \right\|$ is the distance in colour space between the colour vectors $\vec{c}_i$ and $\vec{c}_j$ of the pixels *i* and *j*, and σ_c is defined as 10% of the possible range of distance vector lengths.

As a third criterion *hue* is used. Hue represents the colours of pixels expressed in angles, as part of a cylindrical coordinate system (Burger and Burge, 2006). An object keeps approximately the same hue in a colour image even in the parts darkened by shadows (Perez and Koch, 1994), which means that a different hue almost certainly indicates a different object. The hue weight is set to a fixed value smaller than 1 if the two pixels have a hue difference that is greater than a predefined threshold. A hard threshold is used rather than a slowly decreasing weight function because if the difference is small the weight should not be diminished at all but it should decrease significantly as soon as the difference exceeds a certain value, thus indicating that the pixels belong to different objects. There is certainly some correlation between the colour and the hue criterion, but experiments have shown that the use of both yields the best results. The value of the hue threshold was found by inspecting hue differences between typical road areas with and without shadows, and non-road areas. The value of the hue weight was also found empirically by testing segmentation results with several values. The hue weight is

$$w_{hue} = \begin{cases} w_h & \text{if } d_h > t_{hue} \\ 1 & \text{otherwise} \end{cases}$$

with w_h as the weight assigned if d_h as the hue difference is larger than a threshold t_{hue}.

The fourth criterion is the *normalized difference vegetation index (NDVI)* (Rouse et al., 1973). It is calculated from the infrared and the red channel for each pixel:

$$NDVI = \frac{IR - R}{IR + R}$$

where IR denotes the grey value of the infrared channel of the pixel, and R denotes the grey value of the red channel. The NDVI of a pixel is an indicator for the degree to which the area is covered by vegetation. It is used here for two reasons: first, since roads do not contain vegetation, the segmentation should differentiate clearly between vegetated areas and non-vegetated areas. Second, the difference in the CIR colour space between road areas and areas with weak vegetation (e.g. short grass) is relatively small, even if the colours look distinctly different to a human operator (as noted above, this does not improve by changing the colour space). In order to incorporate the NDVI into the determination of the weights, the NDVI image is calculated. A threshold operation is performed on the NDVI image which separates the image into two NDVI classes: one class containing vegetated regions and the other class containing non-vegetated regions. As the NDVI is dependent on sensor characteristics, relative position of sensor and target, and environmental conditions, the threshold t_{NDVI} must be chosen by the user such that vegetation is clearly separated from non-vegetation. Pixels belonging to the same class with respect to the NDVI are considered similar. This region-based procedure is adopted instead of using the NDVI directly because the goal is to differentiate vegetation from non-vegetation. The direct NDVI difference of, for example, two non-vegetated pixels can be higher than the difference between one vegetated pixel and one non-vegetated pixel. Similar to the hue criterion, the NDVI weight is set to a fixed value. The similarity weight is set to w_{n1} if both pixels belong to the same NDVI region, and it is set to w_{n2} if they do not belong to the same region:

$$w_{ndvi}(i,j) = \begin{cases} w_{n1} & \text{if } r_i = r_j \\ w_{n2} & \text{if } r_i \neq r_j \end{cases} \text{ with } w_{n1} > w_{n2}$$

Here, r_i and r_j are the class labels of the two pixels i and j in the pixel pair.

In order to obtain the final value for the similarity weight, the weights for the four criteria are multiplied:

$$w_{ij} = \begin{cases} \min(1, w_{edge} \cdot w_{colour} \cdot w_{hue} \cdot w_{ndvi}) & \text{if } (i,j) \text{ is active pixel pair} \\ 0 & \text{otherwise} \end{cases}$$

In this manner, weights are calculated for all active pixel pairs *(i, j)* and inserted into the weight matrix.

4.2.2 Normalized Cuts Segmentation

After the calculation of the weight matrix, the normalized cuts algorithm is performed as described in Chapter 3.1. Because of the very large size of the weight matrix, even though it is sparse, the normalized cuts algorithm cannot be computed completely in one step for the images used for the experiments. Therefore, the image is divided into tiles. For each tile, a weight matrix is calculated and the normalized cuts algorithm is applied. The *k*-way cut method (see Chapter 3.1.3) is used to obtain multiple segments; the number of segments for the normalized cuts algorithm is fixed and has to be specified beforehand. After the normalized cuts calculation, the image tiles are joined again and treated as one image in the later steps. Because of the tiling, artificial segment borders occur at the tile borders, which have to be dealt with in the subsequent grouping.

4.3 Grouping

The image segmentation algorithm results in an oversegmentation, which is necessary in order to avoid losing any important road borders. But this enforced oversegmentation often produces segment borders at places where the image information does not justify a separation. Therefore, the initial small segments have to be grouped to larger, more meaningful segments before being further evaluated. The image partitioning which was necessary for the initial segmentation is no longer used, such that the whole image is considered simultaneously for the grouping.

Some criteria used in this step are based on the approach by (Luo and Guo, 2003). They developed a general purpose grouping algorithm, using two sets of criteria: one set concerning properties of a region, such as size, compactness and colour standard deviation, and another set concerning properties of pairs of regions, such as the differences of colour mean values and the edge strength in the border area. In contrast to the general-purpose aim of the grouping algorithm in (Luo and Guo, 2003), the grouping used in this work should be customised for road part extraction. Criteria used by (Luo and Guo, 2003) which are suitable for this aim, particularly the edge strength between regions and the convexity of a region, are adapted from their approach, while other criteria used by them, e.g. the compactness, are disadvantageous for a grouping which is intended for road extraction.

The borders delineating the tiles used for the initial segmentation are treated as segment borders, although with some modulations. An iterative approach is used for grouping the

segments: in each iteration step several pairs of segments are merged, with regard to a number of criteria that are calculated for each neighbouring pair of initial segments.

4.3.1 Grouping Criteria

Some of the criteria for grouping are based on colour and edge features, as in the segmentation, but this time the features of the regions, not those of single pixels, are considered. Additionally, shape features are used, and if a DSM is available, the height difference is also used. Some features are derived from the shared border region, the others relate to the whole regions. Most features are evaluated in a combination with each other, but two features (the NDVI difference and the height difference) are used as "veto features": two regions will not be merged if the criterion for one of these features is not fulfilled. Table 4.1 shows an overview over the features used for grouping.

	shared border regions	whole regions
radiometric	• edge strength	• histogram difference • *NDVI difference*
geometric	• absolute border length • relative border length	• convexity • *DSM height difference (if available)*

Table 4.1. Overview of features used for grouping. Veto features in italics.

Edge strength. The edge strength criterion should prevent regions to be merged if there are strong edges parallel to the border in the shared border region. The shared border region is a narrow band along the shared border (see Figure 4.4) with border width *b*.

Figure 4.4. Two regions with shared border in red and shared border region in cyan.

All edges that lie inside the border region and are parallel to the shared border are used for the edge strength criterion. The mean edge strength is taken from the same edge amplitude image as was used in the initial segmentation. The direction of the edges relative to the border needs to be considered: an edge that runs parallel to the common border divides two segments, while edges that cross the border at a right angle do not contradict a merging. The edge direction for each edge pixel in the image is derived from the gradient. In order to compare the directions, the shared border is divided into parts at points of high curvature by using the split points of the Ramer line approximation algorithm (Ramer, 1972). The average direction of each part is compared to the directions of the edge pixel in its region. Only edges whose directions do not deviate more than 45° from the border direction contribute to the calculation of the mean edge strength; an edge strength of 0 is assumed for all other pixels in the border region. The average edge criterion c_{es} is calculated as

$$c_{es} = \frac{\sum_{i \in P} es_i}{n_{BR}}$$

where es_i is the edge strength of pixel i, which belongs to the set P of all pixels belonging to edges whose directions do not deviate more than 45° from the border direction, and n_{BR} is the number of pixels in the whole border region. The average edge strength criterion is denoted by c_{es}. If only a small part of the border region has a strong edge the separating influence remains small.

Histogram difference. A radiometric criterion relating to the entire segments is the difference of the grey value histograms between the regions in all colour channels. Since a road surface is usually more or less homogeneous, a colour criterion is useful to prevent the merging of two regions that do not have similar colours. Histograms are particularly suited to compare the grey value characteristics of two regions because they represent the complete grey value distribution of the regions (see for example Burger and Burge, 2006). A mean value alone does not have as much significance, and the mean value combined with the standard deviation only describes the distribution completely if it is a normal distribution. The histograms for the two regions are computed and compared separately for each colour channel. The histograms are relative, i.e. they are normalised by the areas of the regions. They are binned with the bin size s_b in order to only capture the significant colour characteristics.

Several methods for histogram comparison are discussed in the literature. An important research field where histogram comparisons are employed is the automatic retrieval of images from databases (see for example Rubner, 1999). No clear preference for one method of histogram comparison can be derived from the literature. In database retrieval robustness against histogram shifts (due to different sensors or different illumination conditions) is critical for the comparison of images. Cross-bin distance measures are therefore preferred. For the grouping of adjacent segments inside one image, however, this kind of robustness is irrelevant or could even have undesired effects. In this work the sum of differences between corresponding bins (L_1 norm of the Minkowski distance) is used to calculate the histogram feature. This measure gives good results when segments of the same image are compared (Rubner et al., 2001):

$$d_{hj} = \sum_i \left| h_{1ji} - h_{2ji} \right|$$

where d_{hj} is the sum of histogram bin differences from channel j, and h_{1ji} and h_{2ji} are the histogram values for each bin i for the regions 1 and 2 in channel j. The histogram difference is calculated for each channel j; the smallest difference is used as criterion for the grouping:

$$c_h = \arg\min_j d_{hj}$$

Experiments have shown that using the smallest histogram distance as criterion yields the best separation between segments that should be merged and segments that should not be merged.

NDVI difference. The NDVI difference is another radiometric criterion relating to the entire regions; it is used to prevent the merging of vegetated regions with non-vegetated regions. The NDVI difference criterion is calculated in a similar way as the NDVI criterion of the initial segmentation, but for a whole region, not individual pixels. The average NDVI for both regions is calculated and then compared to a NDVI threshold. If the average NDVI exceeds the threshold for a region, that region is considered to be vegetated. The NDVI criterion is used to categorically rule out the merging of vegetated regions with non-vegetated regions. It is formulated as follows:

$$c_{NDVI=}\begin{cases}1 & if\ NDVI_1 \neq NDVI_2\\ 0 & if\ NDVI_1 = NDVI_2\end{cases} \qquad NDVI_i = \begin{cases}1 & if\ m_{NDVIi} > t_{NDVI}\\ 0 & if\ m_{NDVIi} \leq t_{NDVI}\end{cases}$$

where m_{NDVIi} is the average NDVI of region i, and t_{NDVI} is the NDVI threshold. This threshold is dependent on the sensor and the conditions during the image acquisition; it is the same as the NDVI threshold used for the NDVI classification for the initial segmentation. The index i takes the values {1;2} for both regions that are compared. If c_{NDVI} =1, the regions will not be merged.

Length of shared border. Geometric criteria of the border region are the *absolute* length and the *relative* length of the shared border. These criteria prevent the merging of segments that only share a border of a few pixels' length, and they prevent the formation of highly irregular segments. An unwanted side effect is the prevention of merging two already long road segments, but this can be corrected in a later step, during the road part extraction. The absolute border length is simply the length of the shared border as shown in Figure 4.4. The relative border length is the ratio of the shared border length to the perimeter of the region with the smaller perimeter.

$$c_{ab} = l_{shared\ border}$$

$$c_{rb} = \frac{l_{shared\ border}}{\min\left(l_{p1}, l_{p2}\right)}$$

The perimeters of the regions 1 and 2 are denoted with l_{p1} and l_{p2}, respectively.

Convexity. Another geometric criterion is the *convexity* of the merged region. A higher convexity is preferred in order to prevent irregular segments, for example segments with long protruding arms. It also prevents the merging of road segments across junctions, which would make a road extraction relying on geometric criteria difficult. The convexity is defined as the ratio of the area of the region to the area of the convex hull of the region. The convexity is 1 if the region is convex; it decreases when the number and size of bulges and holes in the region increases. The convexity criterion is:

$$c_c = \frac{A}{A_{convex\ hull}} \tag{4.2}$$

where A is the area of the merged region and $A_{convex\ hull}$ is the area of the convex hull of the region. The convex hull is the smallest convex region that completely contains the original region.

Height difference. If a DSM is available, the height difference between the regions can be used as additional criterion, in order to prevent merging of regions with different heights, especially road segments and building segments, which can have very similar radiometric properties. The average heights of both segments are derived from the DSM and compared. If the average heights differ by more than 1 m, the regions will not be merged. If the terrain is undulating, a normalised DSM should be used to make sure that road segments along an inclination will still be merged. The use of a normalised DSM means that in terrain consisting of terraces, regions with different terrain heights could be merged. Since this case is usually

far less frequent than sloped roads, it is disregarded in this work. It should be taken into consideration for a more general application.

4.3.2 Combination of Grouping Criteria

All criteria are combined in order to decide whether the segments can be merged. The NDVI and the height difference are used as "veto features": if a segment pair does not fulfil the respective requirements, the segments will not be merged. For the other features it is desirable that they be combined in a way that allows merging if the general impression of the features suggests that the segments belong together. This means that the different features have to be weighed up against each other. For this purpose, a fuzzy control framework (Zadeh, 1965; Klir, 2006) is used.

For each of the five features *edge strength*, *histogram difference, absolute shared border length*, *relative shared border length* and *convexity* a segment pair is assigned to fuzzy sets (Klir, 2006), according to the respective values calculated for the features. The fuzzy sets *ok_for_merging* and *not_ok_for_merging* are used for each feature. For two features (edge strength and histogram difference) the fuzzy set *limited_ok_for_merging* is used additionally. The membership functions for the individual features that assign the segment pairs to the fuzzy sets are shown in Figure 4.5. For the histogram difference, the smallest difference c_h of the three channel differences (see above) is used to calculate the membership degree, but the largest channel difference d_{hmax} and the median channel difference d_{hmed} are considered in the determination of the membership function shapes by checking if

$$d_{hmed} - c_h > 3\left(d_{hmax} - d_{hmed}\right).$$

If this is the case, different membership functions are used such that a high membership degree in *not_ok_for_merging* becomes more likely (Figure 4.5b, dashed lines). The factor 3 is chosen empirically; the distance between the smallest and the median difference should be significantly higher than the distance between the largest and the median difference for the membership function to change.

The interval borders for the fuzzy sets were determined by evaluating samples of region pairs: the samples were checked manually whether they could be merged or not and assigned into groups. This was done for all features separately. As the decision whether two segments should be merged sometimes is ambiguous, four groups were used for the manual assignment: *definitely merge*, *rather merge*, *rather not merge* and *definitely not merge*. After the assignment of the samples into the four groups, the intervals containing incidences of samples in each group were mapped on the feature values (Figure 4.6 shows an example). The interval borders for the fuzzy sets were defined by inspecting the mapping. For example in Figure 4.6, one interval border of full membership to the set *ok_for_merging* was placed at the point where the interval for *rather not merge* begins (point P1). The interval border of no membership was placed at the point where the interval for *definitely merge* ends (point P2).

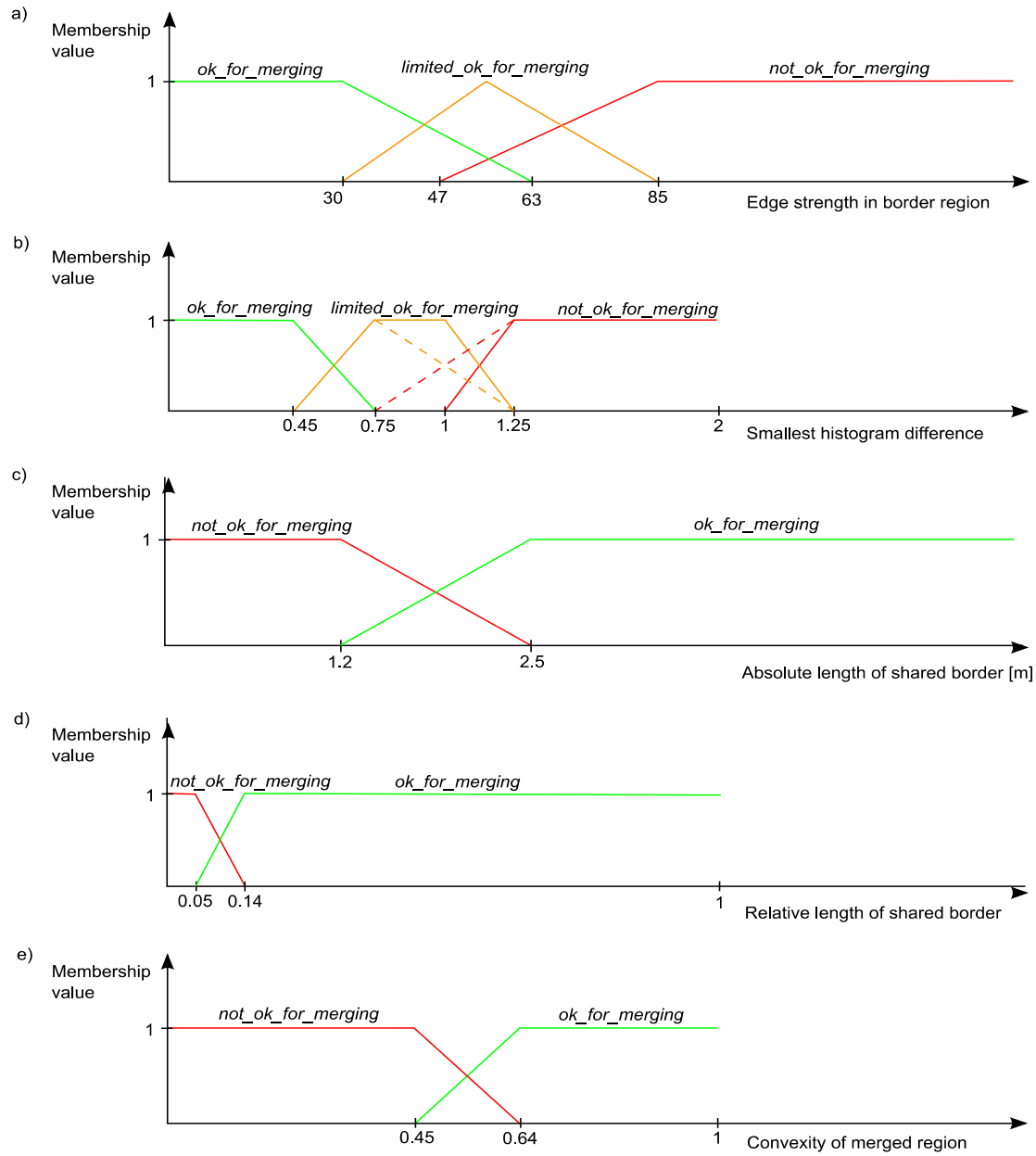


Figure 4.5. Fuzzy membership functions. Membership functions indicating ok_for_merging are shown in green, not_ok_for_merging in red, and limited_ok_for_merging in orange. Dashed lines in b) show alternative functions (see text).

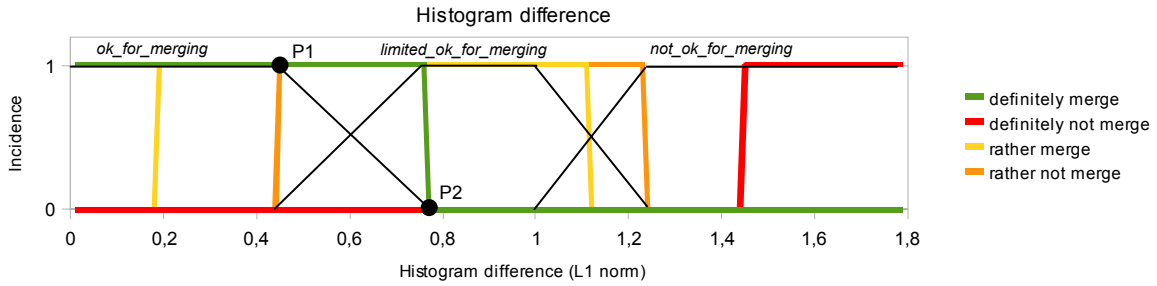


Figure 4.6. Intervals for manual merge decisions, derived from samples and mapped on histogram difference. Thin black lines show derived fuzzy membership functions (Figure 4.5b). Points P1 and P2 mark ends of intervals for ok_for_merging, referenced in the text.

After all feature values for a region pair are assigned to the fuzzy sets with associated membership values, the fuzzy sets are combined using a set of rules in order to arrive at a decision whether the segments can be merged or not. The rules have to take into account that the image had to be divided into tiles for the initial segmentation and that each tile was segmented individually. The segment borders which are formed by the tile borders are artificial and less likely to be good segment borders. Therefore, some rules are different if much of the shared border consists of a division border, in such a way that a merging across a division border is more likely.

The complete set of rules for the merging can be found in the Appendix; only some general considerations will be discussed here. The geometric criteria (border length and convexity) are handled more strictly than the radiometric criteria in order to prevent irregular segment forms. So, for the absolute border length as well as for the convexity, the membership degree of *ok_for_merging* must be higher than that of *not_ok_for_merging*. On the other hand, the edge strength and the histogram difference can be *limited_ok_for_merging*, one of them even *not_ok_for_merging*, if the other features have high membership values in the set *ok_for_merging*. Some conditions are relaxed if the shared border corresponds to a tile border. For example, normally at least one of the features needs to have the membership value 1 for the set *ok_for_merging*, but this is not necessary in case of a division border as shared border. The rules were determined by evaluating samples of region pairs, similar to the interval borders for the fuzzy sets. Characteristic groups of samples were identified, where not all features received the label *ok_for_merging*, but manual inspection suggested that the regions should be merged. The rules were derived from these characteristic groups.

In order to decide which regions should be merged first, an additional *merge value* v_p is calculated for each segment pair p which received a positive merge decision according to the rules. All segment pairs that are candidates for merging are contained in the set M. The merge value is calculated from the original values of the grouping features *convexity*, *histogram difference*, *edge strength* and *relative border length*. The values are normalised based on the maximum value of the respective feature encountered in the current iteration and then added up. The value for the relative border length is additionally weighed according to the current iteration count t such that it weighs less in later iterations. This facilitates the merging of longer road parts in the later stages of the process.

$$v_p = \frac{c_{es_p}}{\max\{c_{es} \mid p \in M\}} + \frac{c_{h_p}}{\max\{c_h \mid p \in M\}} + \frac{c_{rb_p}}{t \cdot \max\{c_{rb} \mid p \in M\}} + \frac{1 - c_{c_p}}{\max\{c_c \mid p \in M\}}$$

The segment pairs are sorted according to their merge values v_p, and the 10% of the segments with the best merge value are merged. If two segment pairs that are candidates for merging share one segment, only the segment pair with the better merge value is merged, the other is ignored. The iteration continues until no more segment pairs receive a positive merge decision.

The result of the segmentation and grouping are segments which are relatively homogeneous in colour, and most segments belonging to road areas are large enough to be evaluated by shape in the next step.

4.4 Road Part Extraction

After grouping, the segments are classified in order to extract road segments. It is often not possible to extract a whole road completely because of disturbances in the appearance of

roads by context objects or varying surface materials. Therefore, the focus here lies on the extraction of reliable *road parts*. The extraction should be reliable enough to generate the network: there must be enough road parts to enable the search for connections, but at the same time the number of false positives should be small enough to allow their elimination in later steps when more global knowledge can be exploited.

For the road part extraction, the compliance of each segment with several criteria is checked. The criteria are based on shape and radiometric characteristics of roads. The following characteristics are used for evaluation:

- Radiometric criteria:
  - Intensity
  - Standard deviation of intensity
  - NDVI
- Geometric criteria:
  - Area and length
  - Elongation, combined with convexity
  - Constant width
  - Width
  - Height

Intensity, standard deviation and NDVI are radiometric criteria. The *intensity* i_{rp} should be higher than a threshold t_{imin} to exclude shadow regions, because shadow regions of buildings often have similar geometric characteristics to road parts, and the calculation of other colour characteristics (such as the NDVI) is less robust for low intensities. Regions with intensities higher than a threshold t_{imax} are excluded as well because in high resolution images, roads usually do not appear nearly white. The permitted intensity range is quite wide, not aiming at specific road colours; it only excludes very high or very low intensity values.

$$t_{imin} \leq i_{rp} \leq t_{imax}$$

The *standard deviation of the intensity* s_{rp} of the road part should not be higher than a threshold t_s, reflecting the fact that a road surface is usually at least piecewise homogeneous.

$$s_{rp} \leq t_s$$

The *average NDVI* $NDVI_{rp}$ is calculated for each segment. For road parts the NDVI should not be higher than the threshold t_{NDVI} because roads in suburban areas are in general not covered by vegetation. The threshold is the same as the NDVI threshold used in segmentation and grouping.

$$NDVI_{rp} \leq t_{NDVI}$$

The other criteria concern the shape of the region. Road parts should have a minimum *length* l_{rp} and a minimum *area* A_{rp} in order to enable a proper evaluation of the other shape criteria. For the estimation of the average width, for example, it must be possible to determine the correct main direction. Therefore, the minimum length threshold t_l should be significantly larger than the average width w_a of a road, and the minimum area threshold t_A corresponds to that of a road part with minimum length and average width. The length is defined here as the distance between those two points on the boundary that lie farthest away from each other,

which is a relatively crude measure but sufficient for checking the minimum length requirement.

$$l_{rp} \geq t_l \quad \text{with } t_l >> w_a$$
$$A_{rp} \geq t_A \quad \text{with } t_A > t_l \cdot w_a$$

The *elongation* e_{rp} of a shape indicates its deviation from a circle. It is given by the ratio of the squared perimeter (the boundary length b_{rp}) and the area A_{rp}, which is high for elongated objects:

$$e_{rp} = \frac{{b_{rp}}^2}{A_{rp}}$$

A road part should have a high elongation. The elongation feature is used in conjunction with the *convexity* c_{rp} of the region. If the region has a high convexity value (cf. Equation 4.2), not less than the threshold t_c, lower values for elongation are permitted (threshold t_{e1}). This allows to include shorter road parts but excludes regions with ragged borders, which also have a relatively high elongation according to the elongation criterion used here. If the convexity value is lower than the threshold t_c, the higher elongation threshold t_{e2} is used.

$$\begin{aligned} e_{rp} &\geq t_{e1} \text{ if } c_{rp} \geq t_c \\ e_{rp} &\geq t_{e2} \text{ if } c_{rp} < t_c \end{aligned} \quad \text{with } t_{e1} < t_{e2}$$

The *width* of a road part should be close to the average road width in the road model, and it should be relatively *constant*. For the calculation of the width, first a centre line is calculated for the region. The centre line calculation is based on a distance transform of the segment borders. First, the two points on the boundary of a segment are found that are farthest away from each other. These points are the preliminary end points of the centre line, an approximation that works well enough for the next steps, even for curved roads up to a 90-degree-bend. The end points divide the boundary into two parts (the left border and the right border), and for each of these two parts a distance transform in the area around them is calculated. The result of the distance transform of a region is a grey value image where the grey value of a pixel encodes its nearest distance to the region (Soille, 1999). The difference of the distance transform images of both border parts is calculated. The centre line is made up of the points where the difference is 0. This centre line calculation is not sufficient for the last parts of the centre line towards the end points. As it is very difficult to determine the correct course of the centre line in these last sections, the course is approximated: the end parts of the centre line are removed and replaced by straight line segments (green lines in Figure 4.7) which have the same direction as the segments of the centre line before the last sections.

Figure 4.7. Centre line determination. Road part border in yellow, centre line in blue with deleted end sections in red and new end sections in green.

The *average width* w_{rp} of the region is calculated from the distances of the centre line to the region borders. Twice the average distance from the borders gives the average width of the region, which should not be too far from the average road width. The minimum width should not be lower than a threshold t_{wmin}. The threshold for the maximum width depends on the length of the segment: if the segment is shorter than a threshold t_{lw}, the width should not be larger than a threshold t_{wmax1}, but if the segment is longer, the width threshold can be more relaxed (t_{wmax2}) because a longer segment is more likely to be a road.

$$\begin{aligned} t_{wmin} \le w_{rp} \le t_{wmax1} \text{ if } l_{rp} \le t_{lw} \\ t_{wmin} \le w_{rp} \le t_{wmax2} \text{ if } l_{rp} > t_{lw} \end{aligned} \quad \text{with } t_{wmax1} < t_{wmax2}$$

The *width constancy* wc_{rp} is measured by the standard deviation s_w of the width divided by the mean width value:

$$wc_{rp} = \frac{s_w^2}{w_{rp}}$$

This value should be lower than a threshold t_{wc1} for elongations up to t_{e2} (see above), although the threshold should not be too strict because of the irregular nature of the segment borders. For regions with an elongation of more than t_{e2} the threshold is more relaxed (t_{wc2}) because the width can vary more for longer segments.

$$\begin{aligned} wc_{rp} \le t_{wc1} \text{ if } e_{rp} \le t_{e2} \\ wc_{rp} \le t_{wc2} \text{ if } e_{rp} > t_{e2} \end{aligned} \quad \text{with } t_{wc1} < t_{wc2}$$

If a DSM is available, the *height* h_{rp} of the segment is used as additional criterion. The average height of a road segment must be close to the ground, i.e. lower than a threshold t_h. The average height is calculated from the normalised DSM.

$$h_{rp} \le t_h$$

All regions are checked for these criteria, and those regions that fulfil all criteria are selected as road parts. Some thresholds for the criteria were derived directly from the road model (such as the minimum area t_A and the minimum length t_l, as described above, as well as the acceptable width range described by the thresholds t_{wmin} and t_{wmax1} or t_{wmax2}); others were found empirically by manual evaluation of a number of segments comprising both acceptable road segments and non-road segments. Non-road segments have a considerably larger variation in the values than road segments, so only the combination of the criteria can yield reliable road segments. The geometric criteria are image invariant, but of course those which have metric units (area, length, width) have to be scaled according to the image resolution. The radiometric criteria are more dependent on sensor characteristics and image acquisition conditions, which is one reason why they have a wider range of acceptable values. For the images used in the tests, only the NDVI threshold had to be adjusted manually for different data sets. An overview over all parameters and dependencies between parameters is presented in Table 4.2.

Parameter		Threshold	Threshold depends on	Origin of threshold
radiometric	Intensity	$t_{imin} - t_{imax}$		empirical
	Standard deviation	$< t_s$		empirical
	NDVI	$< t_{NDVI}$	data set	empirical
geometric	Area	$> t_A$		road model
	Length	$> t_l$		road model
	Elongation	$> t_{e1}$	convexity $> t_c$	empirical
		$> t_{e2}$	convexity $< t_c$	
	Convexity	-	elongation $> t_{e2}$	empirical
		$> t_c$	elongation $< t_{e2}$	
	Width constancy	$< t_{wc1}$	elongation $< t_{e2}$	empirical
		$< t_{wc2}$	elongation $> t_{e2}$	
	Width	$t_{wmin} - t_{wmax1}$	length $< t_{lw}$	road model
		$t_{wmin} - t_{wmax2}$	length $> t_{lw}$	
	Height	$< t_h$		road model

Table 4.2. Overview on parameters for road part extraction. The first column lists the parameters, the second column lists the thresholds for road part extraction. For parameters whose threshold depends on another parameter, both thresholds are listed, and the other parameter is listed in the third column along with the range for which the threshold is valid. The NDVI threshold depends on the data set; in the table the thresholds for the two data sets in the experiments are listed. The fourth column lists the origin of the threshold.

After all road segments are extracted, directly adjacent road segments are checked to determine if they can be merged. This is beneficial because in the previous grouping step adjacent road segments are not always merged due to stricter criteria concerning the convexity and relative border length. After the road part extraction, it is more likely that two remaining adjacent segments belong together. All extracted road parts which share at least 1 m of their borders and whose main directions are sufficiently similar are checked. For the calculation of the main directions only the parts of the roads that are directly adjacent to each other are used, up to a length of 30 m. For this length the roads can be expected to appear relatively straight, which allows curved roads to be merged. If the main directions do not differ by more than 30°, the road parts are merged if the merged road part meets the criteria above described for extracted road parts.

The values for length, elongation, width constancy and deviation from average road width are used to compute a quality measure of the road part. They are mapped onto an interval between 0 and 1 such that values that suggest higher probabilities for road parts are close to 1. For features which do not have an upper limit (elongation and length) all values beyond an upper threshold are mapped to 1. Thresholds used are the same as for the extraction criteria above; additionally, the upper threshold used for the length quality measure is t_{lu}, and additional thresholds for the width quality measure are t_{wl} and t_{wu}. The road width, which must lie between two limits, has an interval around the average road width where values are mapped onto 1, and the mapped values decrease on both sides towards the limits.

$$\text{Quality measure for length: } q_l = \begin{cases} 0 & \text{if } l_{rp} < t_l \\ 1 & \text{if } l_{rp} > t_{lu} \\ 0.5 + \dfrac{0.5 \cdot (l_{rp} - t_l)}{t_{lu} - t_l} & \text{otherwise} \end{cases}$$

$$\text{Quality measure for elongation: } q_e = \begin{cases} 0 & \text{if } e_{rp} < t_{e1} \\ 1 & \text{if } e_{rp} > t_{e2} \\ 0.5 + \dfrac{0.5 \cdot (e_{rp} - t_{e1})}{t_{e2} - t_{e1}} & \text{otherwise} \end{cases}$$

$$\text{Quality measure for width constancy: } q_{wc} = \begin{cases} 0 & \text{if } wc_{rp} > t_{wc2} \\ 1 - wc_{rp} & \text{else} \end{cases}$$

$$\text{Quality measure for width: } q_w = \begin{cases} 0 & \text{if } w_{rp} < t_{wmin} \text{ or } w_{rp} > t_{wmax2} \\ 0.5 + \dfrac{0.5 \cdot (w_{rp} - t_{wmin})}{(t_{wl} - t_{wmin})} & \text{if } t_{wmin} \le w_{rp} \le t_{wl} \\ 1 - \dfrac{0.5 \cdot (w_{rp} - t_{wu})}{t_{wmax2} - t_{wu}} & \text{if } t_{wu} \le w_{rp} \le t_{wmax2} \\ 1 & \text{otherwise} \end{cases}$$

All transformed values are multiplied to obtain a single quality measure q_{rp} for the road part:

$$q_{rp} = q_l \cdot q_e \cdot q_{wc} \cdot q_w$$

As result of the road part extraction, road part hypotheses with associated quality measures are now available for the next step.

4.5 Road Subgraph Generation

After the road part extraction, many roads are covered by one road part, from one junction to the next. But disturbances in the appearance of roads caused by context objects or by a changing appearance of road surfaces, e.g. due to road maintenance, can interfere with the extraction and cause gaps between extracted road parts. In order to bridge the gaps, in this step road parts are connected to their neighbours if they could belong to the same road, forming *road subgraphs* (Figure 4.2).

Two road parts can be assumed to belong to the same road if their geometric relationship indicates that they follow the same course. The features used to decide whether two road parts belong to the same road are:

- distance
- direction difference
- continuation smoothness

The reference points for the computation of the values for these features are the intersection points between the centre line and the road part borders. The connecting line connects those reference points of both road parts which lie closest together (Figure 4.8a).

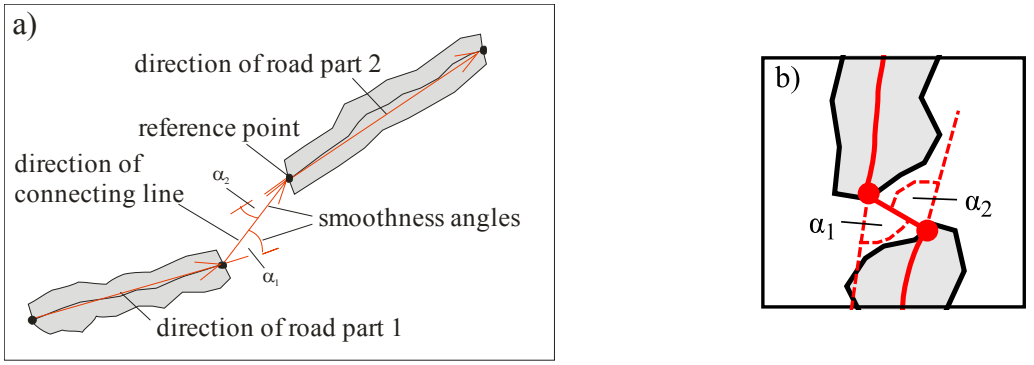


Figure 4.8. Connection between road parts. a) Principle of connection. Road parts and their centre lines displayed in black, end points of centre lines displayed as black dots. Directions of road parts and connecting line indicated by red arrows. b) High lateral offset of two road parts close together; centre lines, connection line and end points of centre lines displayed in red.

For the *distance d* between road parts two aspects are considered: the absolute distance and the relative distance. The *absolute distance* d_a is the distance between the two reference points that lie closest together; it is the same as the length of the connecting line. The *relative distance* d_r is the ratio of the absolute distance and the length of the shorter road part. This measure prevents the connection of very short road parts with a much longer connecting line. Both the absolute distance and the relative distance must be lower than the thresholds t_{da} and t_{dr}, respectively.

The *direction difference* is measured by comparing the directions of the two road parts. The direction of one road part is defined by the vector which connects its two reference points (see Figure 4.8a). The difference is given by the angle between the direction vectors of both road parts; it must be lower than a threshold t_δ.

The *continuation smoothness* is used to describe the property of roads that they usually do not have sudden direction changes in a zigzagging manner. A smooth continuation means that the lateral offset between two connected road parts should be low. To measure the continuation smoothness, the angle between the direction of the road part and the direction of the connecting line is calculated; this is done for both road parts, yielding two angles α_1 and α_2 (Figure 4.8a). Both angles should be lower than the threshold t_α. If the distance between the road parts is shorter than the mean road width of both road parts, the continuation smoothness is disregarded because at close distances the angles depend too much on the exact positions of the reference points: if the reference points of the connecting line are shifted to opposite directions orthogonal to the main road direction, the measured lateral offset will be high although both road parts are aligned well (Figure 4.8b).

The subgraphs are generated iteratively, starting from the road part that received the best quality measure in the road extraction. Potential connection partners for this road part are searched for among the other road parts. Two road parts are connected if the thresholds for the distance, the direction difference and the continuation smoothness are met. The search for neighbouring road parts continues until no other road part can be found that meets the connection conditions. At this point, the subgraph is complete, and the search continues

among the remaining road parts not yet assigned to a subgraph, again starting from the remaining road part with the best quality measure. The generation of subgraphs is continued until all road parts have been examined and assigned to a subgraph. If no neighbouring road parts can be found for a road part, the road part constitutes a subgraph on its own.

One road part can be attached to more than one other road part at the same end point, such that branches in the subgraph can occur (see Figure 4.2). Branches most likely occur in the presence of falsely extracted road parts, but they are still permitted in this stage. This is because the decision which road parts are falsely extracted can be made with greater reliability when the whole subgraph is involved in the decision. In the experiments, most false road parts were actually roofs of buildings, so the occurrence of branches is greatly reduced when a DSM is available.

4.6 Road Subgraph Evaluation

After the subgraph generation, all road parts belong to subgraphs. While some subgraphs can consist of only one road part, others can be composed of several road parts, and some may have branches (Figure 4.2). The ambiguities represented by different branches now have to be resolved. In most cases, branches occur when falsely extracted road parts are present. Connections to falsely extracted road parts can be expected to have a less continuous appearance than connections between road parts that belong to the same road. This is reflected in the properties of the relations between road parts, for example the continuation smoothness or the similarity of colours. Additionally, the space between the road parts must have properties that make the road continuation probable, even if no road part was extracted there. This means that no objects should be present in the gap that would contradict a road hypothesis (e.g. buildings). Other objects (e.g. vehicles) could support a road hypothesis. Following these considerations, two aspects of the connection properties are considered to determine *weights* for the connecting lines: the properties of the *interrelations* of the two connected road parts and the properties of the gap between them, indicated by *context objects*. Both aspects are combined to yield the weights for the connecting lines.

The weights of the connecting lines are used for an optimisation of the whole subgraph, considering all branches together and searching for a global maximum for the subgraph. The optimisation problem is formulated as a linear program. The sum of all weights of the remaining edges should be maximised, subject to the constraints that only one connecting line should depart from each end of a road part. The result after the optimisation is a set of road strings (see Figure 4.2) that do not contain branches. Junctions are not considered in this step; they are reconstructed in the following step (Section 4.7).

In the subsequent sections, first the determination of the interrelation weights (Section 4.6.1) and the context object weights (Section 4.6.2) are described. Both weights are combined in Section 4.6.3, and in Section 4.6.4 the optimisation is described.

4.6.1 Calculation of Interrelation Weights

The interrelation weight w_I of a connection between two road parts is a measure for the plausibility of both road parts belonging to the same road. The features used to calculate the interrelation weight are:

- distance
- direction difference
- continuation smoothness

- road part quality values
- colour difference
- width difference

All features are calculated and converted into weights with a range from 0 to 1, where 1 indicates a good fit of the two road parts.

The first three features (*distance d*, *direction difference* δ_{diff} and *continuation smoothness* (α_1, α_2)) describe the geometric relations of both road parts and were already used for the subgraph generation (cf. section 4.5). The values for these three features are mapped linearly to the interval [0, 1] to derive the feature-specific weights w_{dist}, $w_{\delta diff}$ and w_{cs}. Distance and angle values of zero correspond to the weight 1, and distance and angle values that are the same as the thresholds used for the subgraph generation correspond to the weight 0. For the continuation smoothness value, the larger of both angles is used to calculate the weight. If the distance between the road parts is shorter than the road width, the continuation smoothness value is set to 0.5 because at short distances the relation of the angles to the quality of the connection is low.

$$w_{dist} = 1 - \frac{d}{t_{da}}$$

$$w_{\delta diff} = 1 - \frac{\delta}{t_{\delta}}$$

$$w_{cs} = \begin{cases} 1 - \dfrac{\max(|\alpha_1|, |\alpha_2|)}{t_{\alpha}} & if\ d > \min(w_{rp1}, w_{rp2}) \\ 0.5 & if\ d \le \min(w_{rp1}, w_{rp2}) \end{cases}$$

The *road part quality values* q_{rp1} and q_{rp2} of both road parts are used because falsely extracted road parts usually have lower quality values, so the quality values of both connected road parts should be high. The mean value w_{qrp} of both quality values is used as weight. As both quality values are in the interval [0, 1], the mean will be automatically in the correct range.

$$w_{qrp} = \frac{q_{rp1} + q_{rp2}}{2}$$

The last two features (the *colour difference* and the *width difference*) describe the similarity in appearance of both road parts. The *colour difference* is calculated from the mean values of the colour channels. The mean colour differences between both road parts are calculated for each channel: d_{ir}, d_r and d_g; the largest difference is used to calculate the weight. The difference value is linearly mapped onto the interval [0, 1]; a difference of zero corresponds to the weight 1, a difference that equals the range of allowed mean intensities for road parts, i.e. the difference between the intensity thresholds t_{imax} and t_{imin}, corresponds to the weight 0. The *width difference* d_w is calculated as the difference of the average widths of the road parts. It is also linearly mapped; if the larger width equals twice the smaller width the weight receives the value zero, if both widths are the equal, the weight receives the value 1.

$$w_{col\ diff} = 1 - \frac{\max(|d_{ir}|, |d_r|, |d_g|)}{t_{imax} - t_{imin}}$$

$$w_{width\ diff} = \begin{cases} 1 - \dfrac{d_w}{\min(w_{rp1}, w_{rp2})} & if\ d_w \le \min(w_{rp1}, w_{rp2}) \\ 0 & if\ d_w > \min(w_{rp1}, w_{rp2}) \end{cases}$$

The individual weights from the six features are multiplied to yield the interrelation weight w_I for the connecting line. The interrelation weight lies in the interval [0, 1].

$$w_I = w_{dist} \cdot w_{\delta diff} \cdot w_{cs} \cdot w_{qrp} \cdot w_{col\ diff} \cdot w_{width\ diff}$$

4.6.2 Calculation of Context Object Weights

The second aspect of a connection between two road parts concerns the gap between the road parts. Context objects, if present, are extracted to examine the gap between both road parts, in order to determine whether they contradict or support a road hypothesis in the gap. It can also be assumed that something in the gap must have hindered the road part extraction if the connected road parts really belong together. So, two aspects of context objects are examined:

- their relations with the road part hypothesis in the gap
- the amount of occlusion they cause for the road part hypothesis in the gap.

The context objects considered for the gap examination are vehicles, trees together with their shadows, buildings, vegetated areas and road surface areas. The context objects are extracted automatically, but since their extraction is not the focus of this work, this task will be only described briefly.

For the extraction of *vehicles*, all three channels and the intensity image are first segmented. Two segmentation methods are used and their results are combined: region growing with a very low merging tolerance, and a thresholding operation that yields bright regions. Vehicle regions are selected from the segmented regions according to several shape features such as area, compactness, rectangularity and eccentricity. Two sets of regions are selected: regions which fulfil the shape criteria for whole vehicles, and regions which fulfil the shape criteria for vehicle parts (hood, roof, rear). For the latter, combinations are searched whose relations correspond to the geometric and topological relations between vehicle parts; these combinations are extracted as vehicles in addition to the regions previously selected as whole vehicles. The results from the different channels are merged and redundant extractions are deleted.

For the extraction of *trees* there are three possibilities. If a DSM is available, trees are extracted as high regions with a high NDVI. If no digital surface model is available, the extraction of trees is more difficult and less reliable. Regions with a high compactness[2] and a high NDVI are extracted as tree regions, supported by an extraction of shadows. Shadows are extracted as dark, compact regions. If the position of the shadows relative to the trees is either known or can be reliably determined from the relations between shadows and NDVI regions, then trees can be extracted as compact regions with high NDVI and associated shadow.

[2] Compactness is the ratio of the region area and the area of a circle with the same perimeter. A circle has the highest compactness.

Shadow positions can be determined if enough compact regions of high NDVI (potential trees) have adjacent shadow regions lying in the same direction relative to the potential trees. If shadow regions could not be determined, trees are extracted as regions with high NDVI with stricter shape criteria on compactness and circularity. Only single trees can be extracted in this way.

Buildings are extracted as high and compact objects with low NDVI. No further details are required since the buildings are only needed to assist in the road extraction. Buildings are only extracted if a DSM is available.

Vegetated areas are areas with a high NDVI and a size larger than a minimum size that were not extracted as trees. If a DSM is available, a low height is required as additional criterion. The NDVI threshold for vegetated areas is lower than that for trees, because short grass usually has a lower NDVI than trees.

Road surface areas are extracted as areas with a size larger than a minimum size and the average grey values of the road surface. As the grey values of the road surface can vary considerably, the thresholds for the road surface areas are derived individually for each pair of connected road parts from their grey values. For each channel, the average grey value and the standard deviation are determined from both road parts. The intervals defined by the average grey value ± the standard deviation constitute the accepted grey values of road surface areas within the gap.

The extraction of the context objects is described in detail in (Meyer, 2009), as well as an analysis of the quality of the extraction. In general, the correctness of the extracted context objects is more than 80%; the completeness is lower, lying between 50% and 80%.

A road hypothesis is constructed in the gap between two road parts using the connecting line and the average widths of both road parts. Context objects are extracted in the gap and around the gap in a region of interest whose size depends on the type of object (Figure 4.9).

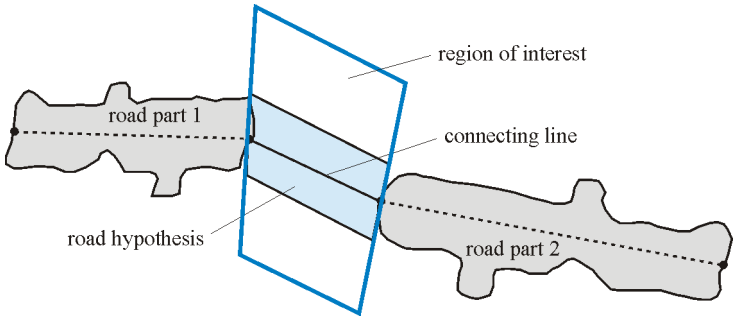


Figure 4.9. Road hypothesis and region of interest for context object extraction. Road hypothesis built around the connecting line displayed in light blue; example for region of interest for context objects indicated by border in blue.

After the extraction of the context objects, a context object weight can be computed for each connecting line. The context objects can contribute to the total weight in two ways: their relation with the road hypothesis in the gap and the amount of occlusion caused by them.

The *context relations* describe the geometric relations of the context objects to the road hypothesis: their locations and orientations relative to the road hypothesis. Based on the location, the context objects can be assigned to the relations *on_road* and *next_to_road*,

depending on the degree of overlap with the road hypothesis. Based on the orientation, the context objects can be assigned to the relations *parallel_to_road*, *perpendicular_to_road* and *diagonal_to_road*, depending on the direction difference between the connecting line and the main axis of the object (see Table 4.3). Orientation assignments only make sense for vehicles and buildings, the other context objects do not have a preferred orientation in their relationships to roads. The combined location and orientation relations lead to several classes for each type of context object. A *relation value* r_j between -0.5 and 0.5 is assigned to each class j, which reflects the strength of contradiction (negative values) or support (positive values) that each relation contributes to the road hypothesis. For very strong contradictions that should not be outweighed by positive evidence the negative value -10 is used. The values were derived empirically from the observed frequencies of the respective relations in the global context of suburban areas, weighted by the estimated importance of the relations in contradicting or supporting road hypotheses.

Vehicles can be assigned to four classes depending on a combination of their location and orientation, none of which contradict a road hypothesis. The relationship class *parallel_on_road* (a combination of *parallel_to_road* and *on_road*) is the most frequent, comprising both moving and parking vehicles, and strongly supports a road hypothesis. The relationship classes *parallel_next_to_road* (a combination of *parallel_to_road* and *next_to_road*) and *perpendicular_next_to_road* (a combination of *perpendicular_to_road* and *next_to_road*) are less frequent; they occur with vehicles parked on separate parking lots or driveways. The relationship class *perpendicular_on_road* (a combination of *perpendicular_to_road* and *on_road*) is the least frequent one; it can only occur with turning vehicles, and consequently only gives weak support to a road hypothesis.

Trees can be assigned to two classes depending on their locations relative to the road hypothesis. The relationship class *next_to_road* is relatively frequent, especially in gaps, because it is one reason for the existence of gaps. Therefore, it gives support to a road hypothesis. As trees can overshadow the roads, a tree is considered as being next to the road if it overlaps the road hypothesis in the gap with up to 70%. The relationship class *on_road* does not occur in the real world and is an indication of a false connection, so it has a negative relation value.

Buildings can be assigned to three classes depending on their location and orientation relative to the road hypothesis. The relationship class *on_road* does not occur in the real world and gives very strong evidence against the road hypothesis. Therefore, it has the high negative value -10, which in practice acts as a veto value against any supporting evidence for the road hypothesis. The relationship class *parallel_next_to_road* is quite frequent, as well as the relationship class *perpendicular_next_to_road*. Both give strong support to the road hypothesis. The relationship class *diagonal_next_to_road* is less frequent, thus giving only weak support.

Vegetated areas can be assigned to two classes depending on their locations. The relationship class *on_road* does not occur in suburban areas and has the high negative value -10, to act as a veto value. The relationship class *next_to_road* gives a weak support to the road hypothesis because it is relatively frequent in suburban areas.

Road surface areas can be assigned to two classes depending on their location. Both the relationship class *on_road* and the relationship class *next_to_road* give supporting evidence for the road extraction, the former more than the latter. In contrast to the other context objects, a road surface area relation is not assigned to a relationship class by a simple overlap

calculation, because a road surface area can stretch beyond the road borders. The relation is assigned to the relationship class *on_road* if more than 50 % of the road hypothesis is overlapped by road surface areas. If more than 30 % of the road surface area is outside the road hypothesis, the relation is assigned to the relationship class *next_to_road*. A road surface area can thus be assigned to both these relationship classes or none of them.

Table 4.3 gives an overview over the relations between context objects and the road hypothesis, the criteria according to which they are assigned and the relation values.

Relation *j*	location (degree of overlap)	orientation (direction difference)	relation value
Vehicle			
parallel_on_road	≥ 50 %	≤ 45°	0.5
parallel_next_to_road	< 50 %	≤ 45°	0.3
perpendicular_on_road	≥ 50 %	> 45°	0.2
perpendicular_next_to_road	< 50 %	> 45°	0.3
Tree			
next_to_road	≤ 70 %	-	0.4
on_road	> 70 %	-	-0.5
Building			
on_road	> 15 %	-	-10
parallel_next_to_road	≤ 15 %	< 15°	0.4
perpendicular_next_to_road	≤ 15 %	> 75°	0.4
diagonal_next_to_road	≤ 15 %	15°-75°	0.1
Vegetated area			
next_to_road	< 25 %	-	0.2
on_road	≥ 25 %	-	-10
Road surface area			
on_road	> 50 % (of road)	-	0.2
next_to_road	< 30 %	-	0.1

Table 4.3. Context object relations. The second and third column give the criteria for assignment to the relations, the fourth column gives the associated relation value.

The relation values r_j for all objects in the gap are combined to a context relation weight $w_{relations}$ as follows:

$$w_{relations} = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \frac{r_j}{i}$$

where k is the total number of relationship classes, n_j is the number of context objects which were assigned to the class j, and r_j is the relationship value assigned to the relationship class j. If more than one context object is assigned to one relationship class, the impact of the next context objects on the total value is decreased (dividing by the current count of context objects i), because the first occurrence of a relation is considered to be more significant for the evaluation of a connection hypothesis than the subsequent ones.

For the second aspect of the context object evaluation the *occlusion* of the road hypothesis by context objects is analysed. An occlusion can cause the road extraction algorithm to fail to extract the road part. For the occlusion analysis the context objects vehicle, tree and shadow are considered, because these are context objects that can occlude parts of the road. A high degree of occlusion by these context objects supports a road hypothesis more than a low degree of occlusion. The second context weight $w_{occlusion}$ is the percentage of the area of the road hypothesis which is covered by the context objects:

$$w_{occlusion} = \frac{A_{context\ objects}}{A_{road\ hypothesis}}$$

Both context weights are added to yield the final weight w_C of the context object evaluation:

$$w_C = w_{relations} + w_{occlusion}$$

4.6.3 Combination of Interrelation and Context Weights

After calculation of both the *interrelation weights* w_I and the *context object weights* w_C, both weights can be combined to yield a *combined weight* w for the connection. The combination of weights follows a set of rules which consider the values of the respective weights as well as the length of the gap:

1. if $w_C < 0$ then $w = 0$
2. else if gap length < average road width then $w = w_I$
3. else if $w_I = 0$ and $w_c \leq 0.5$ then $w = 0$
4. else if gap length > 4 · road width then $w = (w_I + w_C)/2$
5. else $w = (2 \cdot w_I + w_C)/2$.

The first rule reflects the fact that the context object weight will be negative if context objects which strongly contradict a road hypothesis (such as buildings on the road) are present. The second rule prevents the context object weight from having any influence on the combined weight if the gap is very short. For short gaps, the interrelations between two road parts are much more important for the total evaluation of the gap than the context objects. This is especially true for the case where no context objects could be found: in very short gaps the presence of context objects is less probable, and this fact should not decrease the weight. The third rule states that if the interrelation weight is 0 and the context object weight is low, the context object weight is disregarded. The fourth and the fifth rule describe the standard way of combining the two weights by calculating the mean. The fourth rule for longer gap lengths places more importance on the context object weight than the fifth rule for shorter gap lengths. The reason for this is that the interrelations between two road parts are more significant regarding the validity of their connection when they lie closer together. On the other hand, context objects are more important for longer gaps because in order to explain the presence of a long gap in the road extraction, context objects should be present.

The interrelation weight can be used alone if no context object extraction is performed. The context object weight should not be used on its own, especially not for shorter gaps, because the presence or absence of context objects is too unreliable to be used alone. Also, in short gaps, fewer context objects will be found, which leads to lower weights.

4.6.4 Optimisation

In the optimisation of the subgraphs, the subgraphs should be split such that only road strings without branches remain, in order to eliminate connections that are likely to be wrong. This means that some of the connecting lines have to be eliminated. As the weights of the connecting lines describe the degree in which the road parts fit together, the sum of all weights of the remaining edges should be maximised. As no branches are allowed, only one connecting line may be attached to each end of a road part. This optimisation problem can be formulated as a linear program:

$$\text{maximise } \sum_{i=1}^{n} w_i x_i \text{ subject to the constraints for all } k \sum_{j \in E_k} x_j \leq 1, \quad x_j \geq 0 \; \forall \; j$$

where w_i are the weights for the n connecting lines, E_k is the set of edges that belong to the end k of a road part, and x_i are the unknown indicators for the connecting lines. If x_i is 0 after the optimisation, the connecting line will be discarded; if x_i is 1, the connecting line will be kept. In the constraints, the sum of all indicators associated to one end of a road part must not be larger than 1, which means that only one indicator can have the value 1. Thus, the constraints do not allow branches to occur.

The values of the unknowns are restricted to be either 1 or 0, as they are indicators for the connecting lines. As explained in Chapter 3.2.1, the general strategy to solve linear programs does not consider such restrictions, but if the constraint matrix A (Equation 3.9) is totally unimodular and the right side $\vec{b}$ (Equation 3.9) consists of integers, an optimal integer solution to the linear program exists. To recall the conditions for unimodular matrices listed in Chapter 3.2.1, a matrix A is totally unimodular if

1. every column of A has at most two non-zero entries,
2. every entry of A is either 0, 1 or -1,
3. it can be partitioned into two disjoint sets of rows B and C such that if two non-zero entries in a column have the same sign, the row of one is in B and the row of the other is in C, and
4. using the same partition as in 3., if two non-zero entries in a column have opposite signs, both rows are either in B or in C.

The constraint matrix in this case corresponds to the incidence matrix of the graph of the connecting lines. Each row corresponds to one node (end of a road part), and each column corresponds to one connecting line. Condition 1 is fulfilled because each connection line can only be attached to two nodes. Condition 2 is fulfilled because all entries are either 1 or 0, depending on whether the connecting line of the current column is connected to the node of the current row or not. Condition 4 is irrelevant because there are no entries with negative signs. For condition 3 to be fulfilled, it must be possible to partition all nodes of the graph into two disjoint sets such that the end points of each connecting line belong to different sets, i.e. the graph of the connecting lines must be *bipartite*. The graph is bipartite because of the conditions that have to be fulfilled in the road subgraph generation (cf. Section 4.5). There are limits for both the direction difference of two connected road parts and the length of the connecting line. Therefore, all road parts in a subgraph have similar orientations, at least locally, i.e. within the neighbourhood of one connecting line. Thus, the nodes of the subgraph can be partitioned into two parts: one part containing the nodes where the direction vectors of the connecting lines start and one part containing the nodes where the direction vectors of the connecting lines end (Figure 4.10). Each connecting line connects one node from one part with one node from the other part. So, condition 3 is fulfilled, and the constraint matrix is a

totally unimodular matrix. On the right side all entries are 1. Thus, all conditions are fulfilled to solve this optimisation problem as a standard linear program yielding integer results.

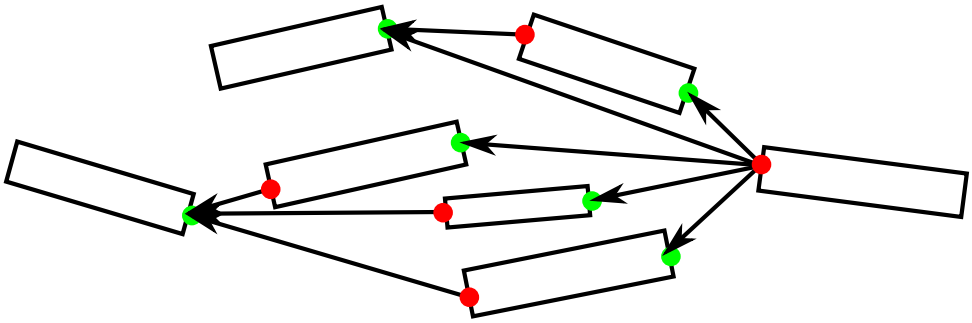


Figure 4.10. Partition of the ends of road parts into two disjoint sets of nodes based on the directions of the connecting lines; the set of nodes where the connecting lines start is displayed in red, the other set in green.

The optimisation is carried out with the revised simplex method in the implementation of the open source linear program solver lp_solve 5.5 (Lpsolve, 2010). The result is an indicator value x_i for every connecting line i. All connecting lines with the indicator value $x_i = 0$ are discarded, the rest are kept. The final result of the subgraph evaluation are road strings (without branches) which represent roads.

4.7 Network Generation

The last stage of the road network extraction is the generation of the road network. The result of the road subgraph evaluation is a set of road strings, each consisting of one or more connected road parts. The road strings represent single roads, which now have to be connected to a road network. For a more suitable representation, first the centre lines of the road parts are merged with the connecting lines and then their shape is approximated by polygons. After that, every road string is represented by a centre line. Before the search for junctions starts, some false positives among the extracted roads can be eliminated. Junction connections are then searched for at the ends of roads. They are checked using context objects, and the road network is formed. The road network can consist of one or more connected components. In a final step, the significance of each connected component (measured by its length) is determined; insignificant connected components are eliminated.

4.7.1 Polygon Approximation and Determination of Average Width

In this stage, a representation of the road strings by their centre lines and average widths is derived. The centre line of a road string is composed from the centre lines of the individual road parts and the connecting lines between them, and then approximated by a polygon. The polygon approximation is an iterative process that starts with a straight line between the end points of the centre line. The average distance of the approximation to the original centre line is measured as follows: the area enclosed by the original centre line and the approximation is determined and divided by the length of the approximation (this corresponds to the mean value theorem for integration). If the average distance is higher than a threshold t_{pa}, the original centre line is divided into two parts of equal length and a new vertex is inserted, such that the approximation now consists of two lines. In each iteration cycle, the number n of parts of equal length into which the original line is divided increases by one, until the threshold condition is fulfilled for each part of the line (Figure 4.11). This way of inserting

new vertices at equidistant intervals - instead of inserting them at the location of the highest distance between the original and the approximation (e.g. Ramer, 1972) - is chosen because in experiments it proved to be a better way of eliminating unwanted bends in the centre line.

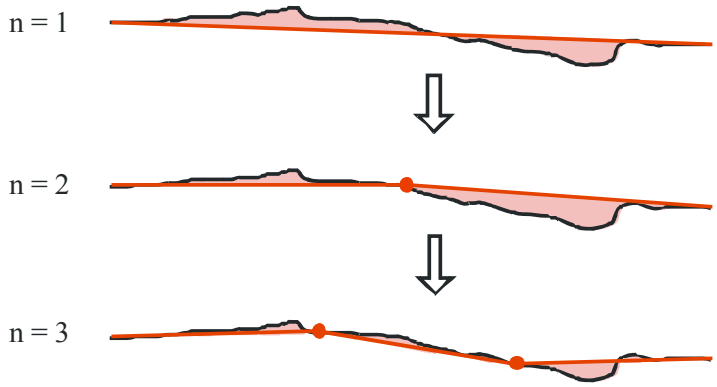


Figure 4.11. Example for vectorisation. Original centre line in black, approximation in red, area used for estimating average distance in light red. Vertices are shown as red circles. In the first iteration, the centre line is approximated by one straight line (n = 1). In the second iteration, the centre line is approximated by two straight lines (n = 2) with a vertex placed in the centre of the original line. In the third iteration, the centre line is approximated by three straight lines, the vertices are evenly distributed.

The average width of the road string is determined by calculating the mean of the average widths of the individual road parts, weighted by their lengths. A quality measure q_r for the road is derived from the quality measures of the road parts in the same way. A region-based representation of the road can be derived from the centre line and the average width by forming a ribbon of the average width along the centre line.

4.7.2 Elimination of Incorrect Road Hypotheses

In order to eliminate as many false roads as possible, pairs of parallel roads that lie close together are searched for. All road pairs whose minimum distance is less than the length of a typical block of houses t_{pd} are examined. The straight polygon edges by which the road was approximated are compared individually to find parallel pairs. A parallel pair in this case is a pair of polygon edges from both roads with an orientation difference ω_p of less than a threshold t_{por}, a minimum distance d_p of less than t_{pd} and an overlap o_p of at least a threshold of t_{pov} (Figure 4.12). Here, the overlap refers to the projection of one polygon edge onto the other. If such a parallel pair is found for two roads, only one road is kept. Which road is kept depends on the lengths and the quality measures of both roads. If one road is more than p_{ld}% longer than the other, the longer road is kept. The reason for this is the assumption that roads are among the longest structures in suburban areas; a longer structure is therefore more likely to be a road. If both roads are of approximately the same length, the measure of length is not as relevant; in this case the road with the better quality measure is kept.

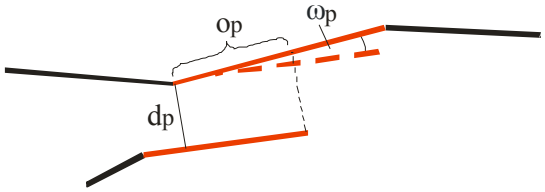


Figure 4.12. Pair of parallel straight edges; d_p is the distance between the edges, o_p is the overlap, ω_p is the orientation difference.

This step is particularly important if no DSM is available as it very efficiently eliminates false extractions that lie on building roofs parallel to extracted roads.

4.7.3 Search for Junctions

Junction connections are searched for among the roads remaining after the previous step. From the ends of each road centre line, connections to other road centre lines are examined within a search region with a certain search radius. The search radius depends on the quality measure of the road: a road with a good quality measure has a large search radius. To determine the search radius, the quality measure is mapped to an interval between the smallest and the largest accepted search radius by linear interpolation: quality measures between t_{qmin} and t_{qmax} are mapped to a search radius interval between r_{min} and r_{max}. For smaller quality measures, the search radius is r_{min}, for larger quality measures, the search radius is r_{max}. If the search radius thus determined is longer than the length of the road, it is set to the road length.

Three types of connections between two disjoint roads are possible, depending on whether both roads are collinear or not (Figure 4.13 a-c). If another road centre line is found inside the search region around one road end, it is checked whether a connection between both roads can be created. A fourth type of connection between two road centre lines occurs if two roads intersect directly (Figure 4.13 d).

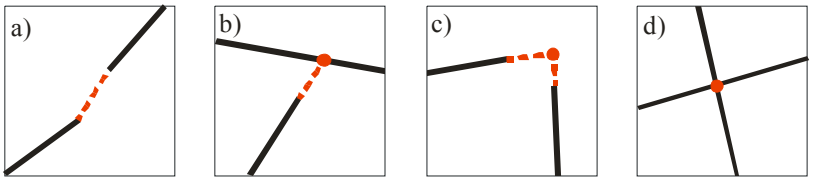


Figure 4.13. Connection types between two roads: a) connection between two nearly parallel roads; b) connection by single extension between two non-parallel roads; c) connection by double extension; d) intersection of two roads. Roads are displayed as black lines, connections as dashed red lines, junction points as red dots.

If both roads are collinear, i.e. have a direction difference smaller than a threshold $t_{\delta c}$, they are connected between their end points (Figure 4.13 a) if the end point of the second road lies inside the search region of the first or vice versa. The two straight polygon sections closest to each other from both roads are used to calculate the orientation difference. For the connection to be accepted, the direction difference between each road and the connection should be lower than a threshold t_{ac}. Additionally, the connection is verified using context objects.

The context objects which were extracted for the subgraph evaluation are used again here for the verification of junctions. Only context objects that can contradict a road hypothesis (trees, vegetated areas and buildings) are considered. For the check, a connection region is defined around the line of the connection, with the mean width of both roads. If a context object covers the connection region to a significant degree, the connection is rejected. For *trees*, the connection is rejected if more than 80 % of the area of a tree covers the connection region. For *vegetated areas*, the connection is rejected if more than 20 % of the connection region is covered, or if the vegetated area crosses the connection region. That is for example the case when a false connection crosses a grass verge. For *buildings*, the connection is rejected if more than 10 % of the connection region is covered, or if the connection line crosses a building. A connection is also rejected if it crosses another road. If no context objects are used for the road extraction, the junction connections are verified by computing the average NDVI

of the connection region. If the average NDVI is lower than the threshold t_{NDVI}, the connection is accepted.

If the roads are not collinear, there are two possibilities for connecting both roads. The first possibility is an extension of the road R_1 to a point on the other road R_2, following the direction of the polygon edge to which the end point belongs (Figure 4.13 b). This connection C_1 is verified using context objects in the way just described. Additionally, if the connection is not rejected, a connection verification value v_{conn} is calculated from the three context object classes as follows:

- Trees:

$$v_{conn\_trees} = \begin{cases} 0.5 & \text{if more than 50\% of the tree area covers the connection region} \\ 1 & \text{otherwise} \end{cases}$$

- Vegetated areas:

$$v_{conn\_veg} = \begin{cases} 0.5 & \text{if more than 2\% of the connection region is covered by veg. areas} \\ 1 & \text{otherwise} \end{cases}$$

- Buildings:

$$v_{conn\_buildings} = \begin{cases} 0.5 & \text{if more than 10\% of a building lies inside the connection region} \\ 1 & \text{otherwise} \end{cases}$$

The three values are combined by calculating the mean:

$$v_{conn} = \frac{v_{conn\_trees} + v_{conn\_veg} + v_{conn\_buildings}}{3}$$

If the connection verification value of C_1 is less than 1, the shortest path between both roads is checked as another connection possibility C_2. If the verification value of C_2 is also less than 1, a better connection is searched for among a number of connections lying between C_1 and C_2. These connections are created by shifting the end point lying on R_2 between the end points of the connections C_1 and C_2 (Figure 4.14 a) in 1m intervals. The connection with the best connection verification value is kept; if several connections have the same connection value, the connection closest to the straight extension C_1 is kept. A junction point is created at the point where the extension from the road R_1 meets the other road R_2. The extension is merged with the road R_1 from which it proceeds; the road R_2 is split at the junction point. If all connections are rejected, the roads remain unconnected.

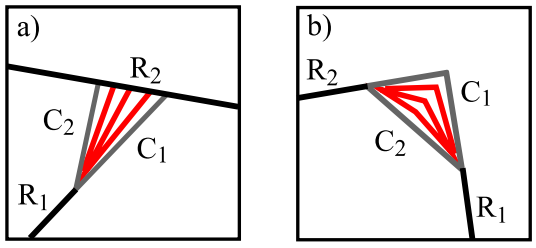


Figure 4.14. Possible connections between two roads: a) single extension from road end to road; b) double extension between two road ends. Roads displayed in black, extremal connections C_1 and C_2 in grey, other possible connections in red.

The second possibility for a connection of non-collinear roads is via the extension of both roads (Figure 4.13 c). The directions of the extensions are the same as the directions of the polygon edges to which the end points belong. The maximum length of the extensions corresponds to the search radius. The connection C_1 is composed of both extensions, and it is

verified using context objects in the way described above. If the connection verification value is less than 1, or if the connection is longer than the search radius, a direct connection C_2 between both roads is checked, as well as a number of connections between C_1 and C_2 (Figure 4.14 b). These additional connections are created by shifting the point where the two extensions meet in 1 m intervals along a line perpendicular to C_2. The connection with the best verification value is kept. If several connections have the same verification value, the connection closest to the original double extension is kept. A junction point is created at the point where both extensions meet, except for a direct extension, where the junction point is created at the centre of the extension line. Both extension lines are merged with the roads from which they proceed. If all connections are rejected, the roads remain unconnected.

Additionally, intersections between two roads are searched for. It is checked whether any two roads intersect (Figure 4.13 d). At the point of an intersection a junction point is created; no other verifications are performed. Both roads are split at the junction point. These intersections can occur where two road parts were connected across a gap that crosses another road.

4.7.4 Final Network Check

The road network now consists of one or more connected components of roads and junction points. Each connected component is examined to determine whether it is significant enough to be kept. Two aspects are considered: the total length of the roads in the connected component and the length ratio of added junction connections to the whole connected component.

The total length of the network should exceed a threshold t_{nl}; shorter isolated connected components are unlikely to belong to the road network and are consequently deleted. An exception is made if at least two end points of the network lie closer to the border of the image than a threshold t_{bd}, because then they could belong to the road network beyond the image border.

The extensions added to the roads during the search for junctions should not account for the majority of the total length of a connected component, because then a large part of the network would consist of relatively weak hypotheses. Therefore, a connected component is deleted if connection extensions contribute to more than 40 % of the total length of the road network.

In a last step, very short dead end roads – dead end roads with a length of less than a threshold t_{dr} – are deleted because it is more likely that they were caused by the junction search process than that they are proper roads. They most often occur when two roads intersect (Figure 4.13d) at a T-junction.

The final revised road network consists of roads, represented by their centre lines and associated road widths, and junctions, represented by the points where the centre lines meet.

5 Experiments

The method presented in this thesis was tested on some image subsets taken from two data sets which show suburban scenes from different places. In this chapter, first the data used in the experiments are described, then all steps from the image segmentation to the road network generation are shown for one relatively small example subset. Finally, a quantitative analysis of the results for all image subsets is presented, with a comparison of the results with and without a DSM. The impact of several features used for the extraction is also examined.

5.1 Data Sets

Two different data sets are used to test the approach. Both consist of aerial orthoimages showing suburban scenes, with associated DSMs. The first data set is a scene from a suburban area of Grangemouth, Scotland; the second data set contains suburban scenes from the region of Vaihingen, Germany.

5.1.1 Grangemouth Data Set

The Grangemouth data set consists of a CIR orthophoto and a DSM. The orthophoto was generated from scanned aerial images (captured by the French ISTAR company in 2000) and has a resolution of 10 cm. The DSM was generated from the images by image matching with the method described in (Gabet et al., 1997); it has a grid width of 20 cm and a height resolution of 10 cm. The DSM has a high quality with regard to elevated objects. Two subsets of the image were used in the tests (Figure 5.1, left). The first subset comprises an area of 562 m x 485 m (5617x4849 pixels) and contains a road network with a total length of approximately 3.75 km and 20 junctions. The second subset comprises an area of 500 m x 500 m (4998x4998 pixels) and contains a road network with a total length of 3.8 km and 22 junctions.

5.1.2 Vaihingen Data Set

The Vaihingen data set consists of several digital aerial CIR images captured with 11 bit and reduced to 8 bit. They were captured with a DMC camera, covering the town of Vaihingen an der Enz. The DSM was derived from LIDAR height data (approx. 5 points/m²), acquired four weeks later than the images. Images and LIDAR data were acquired by the German Association for Photogrammetry and Remote Sensing (DGPF) as test data set for camera evaluation (Cramer, 2010; see also http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html). Orthophotos with a resolution of 8 cm were generated from two images. The DSM has a grid width of 0.25 cm and a height resolution of 10 cm. Two subsets from the orthophotos were used in the tests (Figure 5.1, right). The first subset comprises an area of 394 m x 357 m (4929x4465 pixels) and contains a road network with a total length of 2.41 km and 16 junctions. The second subset comprises an area of 400 m x 400 m (4993x4993 pixels) and contains a road network with a total length of 2.2 km and 13 junctions.

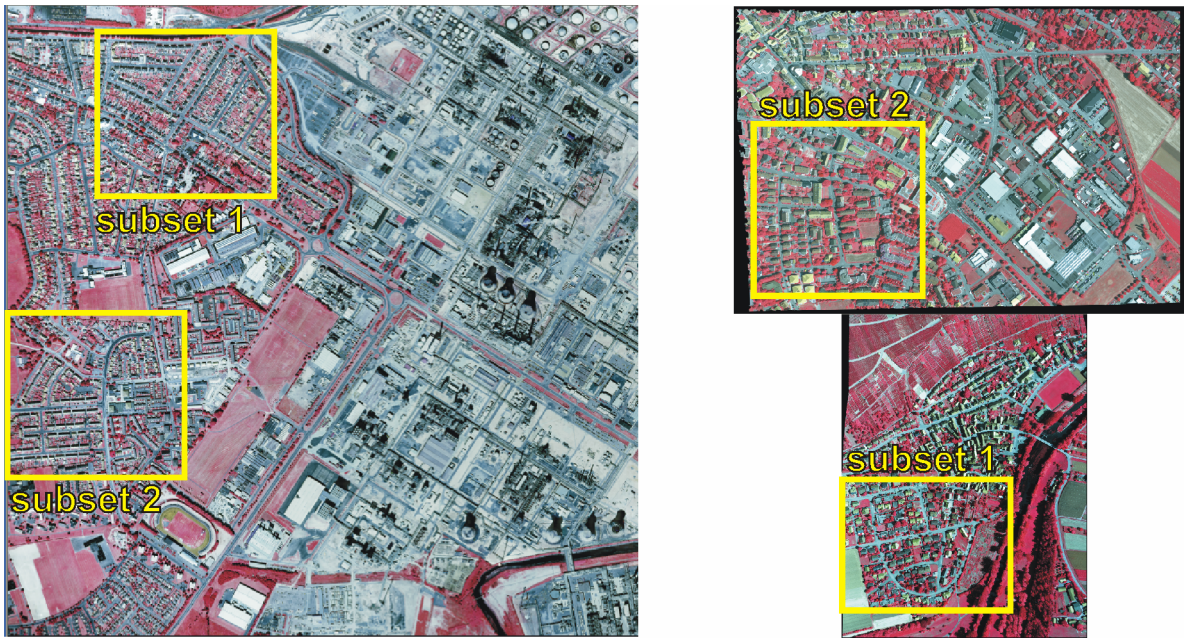


Figure 5.1. Overview of image data. Left: Grangemouth; right: Vaihingen. Image subsets indicated by yellow frames.

5.2 Steps of Road Network Extraction – Example

The extraction of a road network is described here for a small image subset in all steps from the segmentation to the road network. The original image subset is shown in Figure 5.2, together with the DSM. In Figure 5.3, the result of the normalized cuts segmentation is shown. The image is divided such that the tiles have a size of approximately 200 by 200 pixels. Keeping in mind that an oversegmentation is desired to separate road regions and non-road regions as completely as possible, the number k of segments for each partition is set to 20. This number was empirically found to yield good results for most of the test images. The other parameters used for the normalized cuts segmentation are listed in Table 5.1; for a description of the parameters refer to Chapter 4.2. The parameters listed in this table and the following tables are used without change for all data sets that were examined. The only exception is the NDVI threshold t_{NDVI}, whose value is different for the Grangemouth and the Vaihingen subsets.

The initial segmentation yields small, compact segments. The grid structure which can be seen in the segment borders comes from the tiling of the image which is necessary for computational reasons. In the subsequent grouping (Figure 5.4), nearly all of these artificial borders vanish. After grouping, the regions are larger, and most road parts belong to elongated regions. Some parameters used for the grouping are listed in Table 5.2. The parameters used for the thresholds of the fuzzy membership functions are shown in Chapter 4.3.2.

Figure 5.2. Example subset, original image. Left: aerial image, right: DSM.

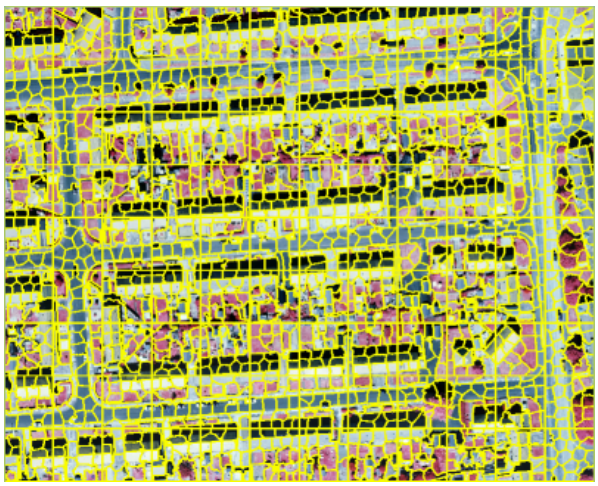


Figure 5.3. Normalized cuts segmentation. Segment borders in yellow.

Figure 5.4. Grouped segments. Segment borders in yellow.

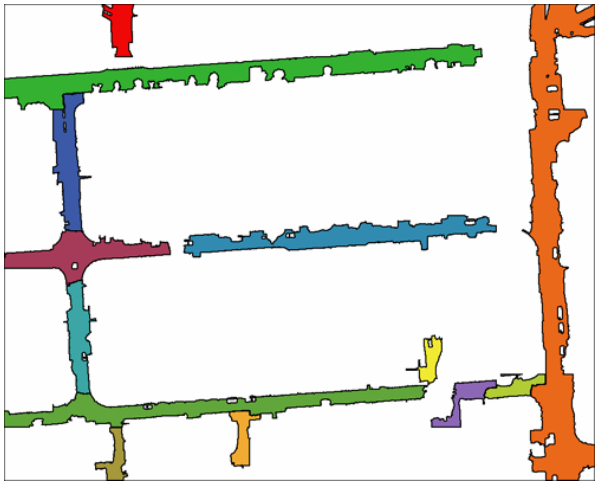


Figure 5.5. Extracted road parts.

Parameters for segmentation		
n	max. neighbourhood distance	10 px
p_n	percent of neighbourhood weights computed	50%
σ_{LoG}	standard deviation of Gaussian	2
w_h	hue weight if over threshold	0.01
t_{hue}	hue threshold	40°
w_{n1}	NDVI weight same region	2
w_{n2}	NDVI weight different regions	0.01
t_{NDVI}	NDVI threshold (Grangemouth)	0
	NDVI threshold (Vaihingen)	0.2

Table 5.1 Parameters for segmentation.

Parameters for grouping		
b	width of border region	7 px
s_b	bin size of histograms	10

Table 5.2 Parameters for grouping.

After grouping, road parts are extracted from the segments. The road parts can be seen in Figure 5.5; most road areas in this image could be extracted as road parts. There are no false extractions apart from some small parking lots which cannot be distinguished from road parts. The parameters used for the road part extraction are listed in Table 5.3.

Parameters for road part extraction					
t_{imin}	minimum intensity threshold	40	t_{wmax1}	maximum width threshold 1	12 m
t_{imax}	maximum intensity threshold	210	t_{wmax2}	maximum width threshold 2	14 m
t_s	maximum standard deviation threshold	50	t_{lw}	length threshold for max. width	50 m
w_a	average road width	6.5 m	t_{wc1}	width constancy threshold 1	0.6
t_l	length threshold	19 m	t_{wc2}	width constancy threshold 2	0.7
t_A	area threshold	130 m²	t_h	height threshold	1.5 m
t_{e1}	elongation threshold 1	34	t_{lu}	upper length threshold for quality measure	100 m
t_{e2}	elongation threshold 2	80	t_{wl}	lower width threshold for quality measure	5.5 m
t_c	convexity threshold	0.53	t_{wu}	upper width threshold for quality measure	8 m
t_{wmin}	minimum width threshold	4.4 m			

Table 5.3. Parameters for road part extraction.

Road parts which could belong to the same road are connected to form subgraphs (Figure 5.6). Table 5.4 shows the parameters for the road subgraph generation. The subgraphs in green, yellow and cyan consist of two road parts. In this example, there are no branching subgraphs, so the subsequent subgraph evaluation just checks if the connection is valid. In order to show branching subgraphs, the steps of the algorithm up to the subgraph evaluation were repeated without using the DSM. Figure 5.7 shows the results for the subgraph generation without the DSM. Two subgraphs (the yellow subgraph and the green subgraph) contain branches. The subsequent subgraph evaluation decides on which branches to keep; for this purpose context objects are extracted. The results of the context object extraction are shown in Figure 5.8. The subgraphs of the example without DSM after the subgraph evaluation are shown in Figure 5.9. The new subgraphs do not contain any branches; the remaining connections between correctly extracted road parts follow the course of the roads.

Parameters for road subgraph generation		
t_{da}	absolute distance threshold	50 m
t_{dr}	relative distance threshold	1
t_δ	direction difference threshold	40°
t_α	smoothness angle threshold	40°

Table 5.4. Parameters for road subgraph generation.

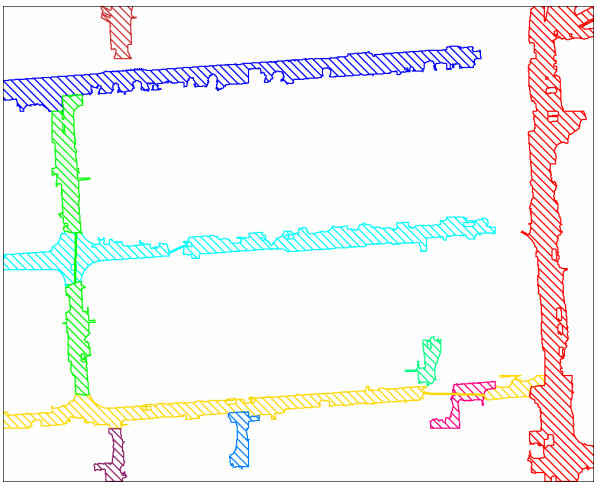


Figure 5.6. Road subgraphs, with DSM. Connected road parts of the same colour belong to the same subgraph.

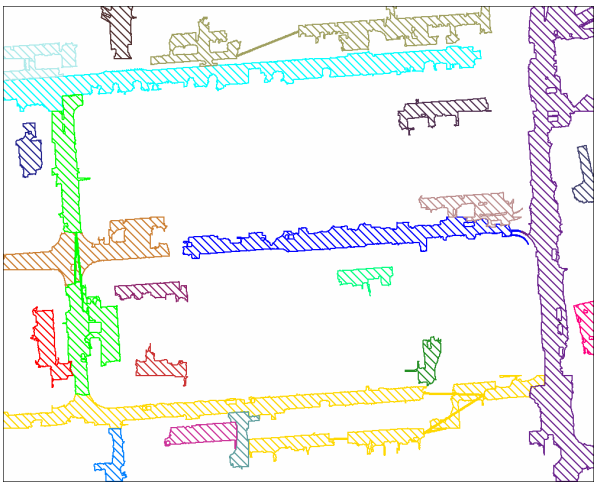


Figure 5.7. Road subgraphs, without DSM. Connected road parts of the same colour belong to the same subgraph.

Figure 5.8. Context objects. Green: vegetated areas; dark green: trees; yellow: vehicles; red: buildings.

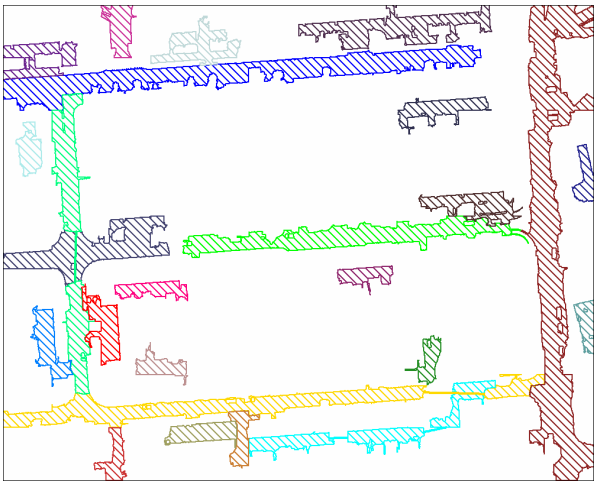


Figure 5.9. Road subgraphs without DSM, after elimination of branches. Connected road parts of the same colour belong to the same subgraph.

The centre lines of the road parts and the connections between the road parts are joined and then approximated to yield relatively straight centre lines. Figure 5.10 shows the approximated centre lines and the boundaries derived from the average widths of the roads. After the approximation, parallel roads are eliminated. Connections between the roads are searched in the last step to complete the road network (Figure 5.11). The parameters for the polygon approximation, the elimination of parallel roads and the network generation are listed in Table 5.5.

Parameters for polygon approximation, elimination of parallel roads and network generation					
t_{pa}	polygon approximation threshold	1.5 m	t_{qmin}	minimum quality measure threshold	0.01
t_{pd}	parallel distance threshold	30 m	t_{qmax}	maximum quality measure threshold	0.9
t_{por}	parallel orientation threshold	30°	$t_{\delta c}$	direction difference threshold	30°
t_{pov}	parallel overlap threshold	5 m	$t_{\alpha c}$	continuation smoothness threshold	45°
p_{ld}	percent of length of longer road	20%	t_{nl}	threshold for length of connected component	100 m
r_{min}	minimum search radius	5 m	t_{bd}	threshold for distance to border	10 m
r_{max}	maximum search radius	50 m	t_{dr}	threshold for length of dead end roads	10 m

Table 5.5. Parameters for polygon approximation, elimination of parallel roads and network generation.

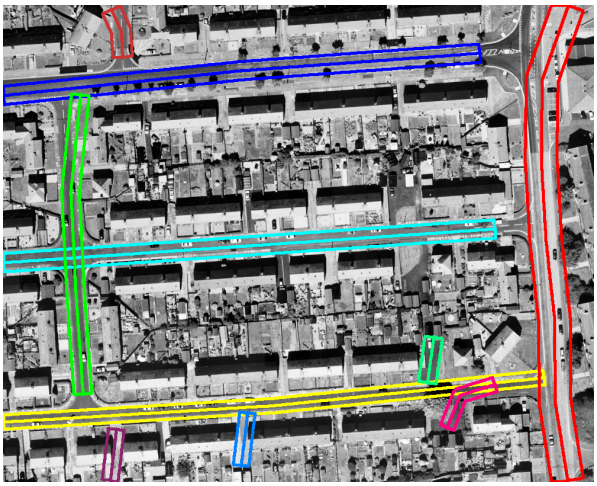


Figure 5.10. Approximated centre lines and widths.

Figure 5.12. Road network. Road centre lines in yellow, junction points in red.

5.3 Results and Quantitative Analysis

For the four subsets described in 5.1, a quantitative analysis of the results was performed. The extracted networks were compared to reference data, based on their centre lines. The reference data were generated by extracting road centre lines manually from the orthoimages. Each reference data set contains the road centre lines as lines and the junctions as points where several lines meet. The manually extracted road centre lines can be expected to lie within a distance of ≤ 1 m from the true centre line.

Several quality measures were used for the quantitative analysis, following the quality measures defined in (Wiedemann et al., 1998; Wiedemann and Ebner, 2000). The quality measures used are described in Section 5.3.1. The quantitative analysis for the results of the image subsets is presented in Section 5.3.2, and the influence of some of the features used in the extraction is examined in Section 5.3.3.

5.3.1 Measures for Quantitative Analysis

The quality measures to be used for the quantitative analysis can be arranged in three groups. The first group measures the quality of the road extraction itself, the second group measures the quality of the topology of the extracted network, and the third group measures the quality of the junction extraction. The first group is the most important one, because the results of the second and third group depend on the road extraction – with bad results in the road extraction quality measures, good results in the other groups cannot be expected. Additionally, the measures of the first group are the most important ones for a comparison of the results, as they are the measures given most often as results in the literature.

The first group of quality measures consists of the completeness and the correctness of the detected road centre lines, as well as the geometric accuracy of correctly extracted roads. In order to determine the *completeness*, a buffer is constructed around each extracted road centre line, and the length l_{Comp} of all reference line segments inside the buffer is determined. The buffer width is set to ±5 m according to the typical road width in our test areas. The completeness is defined as the percentage of the length l_{Comp} of reference roads inside the buffer compared to the overall length l_{Ref} of the reference road network. In order to determine the *correctness*, the buffer is constructed around the reference roads, and the length l_{Corr} of all extracted road centrelines inside the buffer is determined. The correctness is the percentage of the length l_{Corr} of extracted roads inside the buffer compared to the overall length l_{Extr} of the extracted road network:

$$completeness = \frac{l_{comp}}{l_{ref}}$$

$$correctness = \frac{l_{corr}}{l_{extr}}$$

The geometric accuracy of the extracted roads is measured by the *RMS error of the distance* RMS_D. It is computed from the shortest distances d_{re} of a large number n_p of points (in intervals of 10 cm) on the correctly extracted roads from their nearest reference road:

$$RMS_D = \sqrt{\frac{\sum d_{re}^2}{n_p}}$$

A high geometric accuracy of the road centre lines is not expected. There are several reasons for this. First, the approximation of the centre lines is rather coarse in order to remove unwanted bends; the focus does not lie on an accurate geometric reconstruction of the centre line. Second, the buffer for the completeness and correctness check is relatively wide in order to consider all extracted centre lines within the road areas, and third, the accuracy of the manually extracted reference centre lines is limited. Selecting a smaller buffer width would result in better values for RMS_D at the cost of lower completeness and correctness values.

The second group of quality measures, related to the topology of the extracted road network, consists of the *mean detour factor*, the *topological completeness* and the *topological correctness*. For the calculation of these measures, two paired sets of nodes were defined in both the extracted network and the reference network. For the calculation of the topological completeness, the primary set of nodes R consists of points distributed along the reference network with a distance of 20 m, including the junction points. The associated set of nodes

E_{AR} consists of those points in the extracted network which are nearest to the nodes in R, with a maximum allowed distance of 15 m. The node sets for the assessment of the topological correctness are found analogously, with the primary set E derived from the extracted network and the associated set R_{AE} derived from the reference network.

The *mean detour factor* F_D is the factor by which the average path length between two points in the extracted network is longer than in the reference:

$$F_D = \frac{1}{n_P} \sum_{\{i,j\} \in P} \frac{D^E_{i,j}}{D^R_{i,j}} \tag{5.1}$$

In Equation 5.1, P is the set of all node pairs i, j that are connected in both the extracted network and the reference network, taken from the sets R and E_{AR}, and n_P is the number of these pairs, whereas $D^E_{i,j}$ and $D^R_{i,j}$ are the shortest paths between the nodes i and j in the extraction results and the reference, respectively. The optimal value of the detour factor is 1; the value increases if there are detours in the extracted road network compared to the reference network. An analogously defined shortcut factor can be computed, too, but as the shortcut factor was better than 0.98 in most cases, and never below 0.97, it is not listed for the individual data sets. The *topological completeness* measures the percentage of connections in the reference network that also exist in the extracted network; it is calculated using the sets R and E_{AR}. All connections in R that are also connections in E_{AR} are counted as matched connections and compared to the total number of connections in R. Similarly, the *topological correctness* is the percentage of connections in the extracted network that also exist in the reference network, calculated using the sets E and R_{AE}. With n_i denoting the number of the respective connections, these quality measures are determined from:

$$top.\,completeness = \frac{n_{matched\ connections}}{n_{reference\ connections}} \tag{5.2}$$

$$top.\,correctness = \frac{n_{matched\ connections}}{n_{extracted\ connections}} \tag{5.3}$$

The last group of quality measures is related to the extracted junctions. The junction completeness is the percentage of junctions in the reference that could be assigned to extracted junctions based on a distance criterion defined similarly to the evaluation of the road centre lines. It is calculated analogously to the topological completeness (Equation 5.2), using the respective numbers of junctions. The correctness of the junction extraction is the percentage of extracted junctions that could be assigned to junctions in the reference and is calculated in a similar manner to the topological correctness (Equation 5.3).

$$junction\ completeness = \frac{n_{matched\ junctions}}{n_{reference\ junctions}}$$

$$junction\ correctness = \frac{n_{matched\ junctions}}{n_{extracted\ junctions}}$$

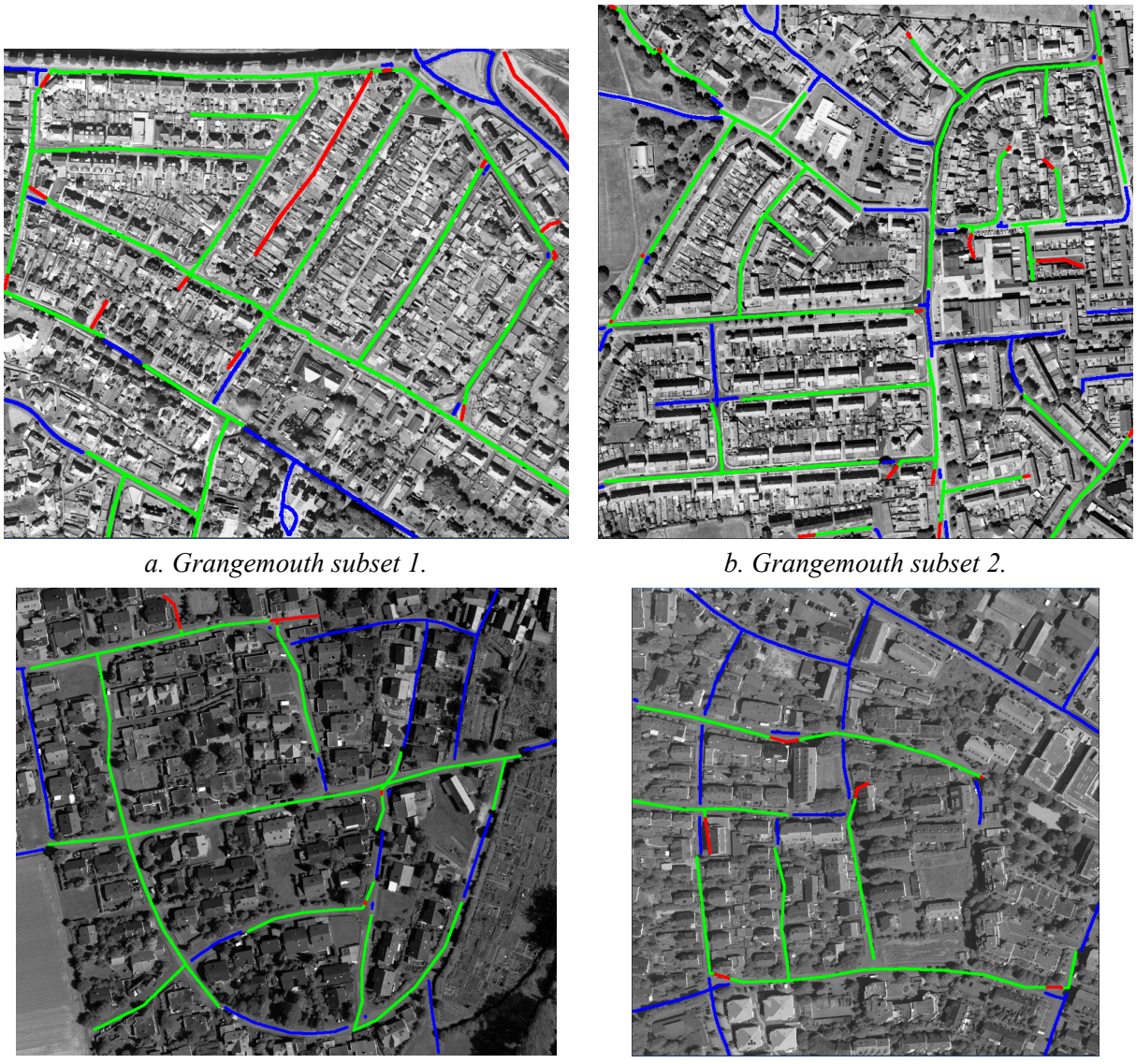
The buffer for the junction evaluation is larger than for the road evaluation because the extraction of the junction points is not as accurate; the junctions are not directly extracted from the image, but often reconstructed from extensions of the ends of the road. Inaccuracies

of the road centre lines are usually largest at the ends of roads, which results in an even higher offset at the ends of the extensions. Therefore, the buffer radius is set to 15 m for the junction evaluation. The geometric accuracy of the matched junctions is further evaluated by calculating the *RMS difference of the junctions* RMS_J from the sum of the distances between matched junctions d_{jm}. The number of matched junctions is denoted as n_{jm}:

$$RMS_j = \sqrt{\frac{\sum d_{jm}^2}{n_{jm}}}$$

5.3.2 Quantitative Analysis of Results

The results of the extraction for all four subsets are shown in Figure 5.13, together with the comparison with ground truth. Correctly extracted roads (true positives) are shown in green, missed roads (false negatives) in blue, and erroneously extracted roads (false positives) in red. The results of the quantitative analysis are shown in Tables 5.6 to 5.8. All results were obtained from extractions including the DSMs.



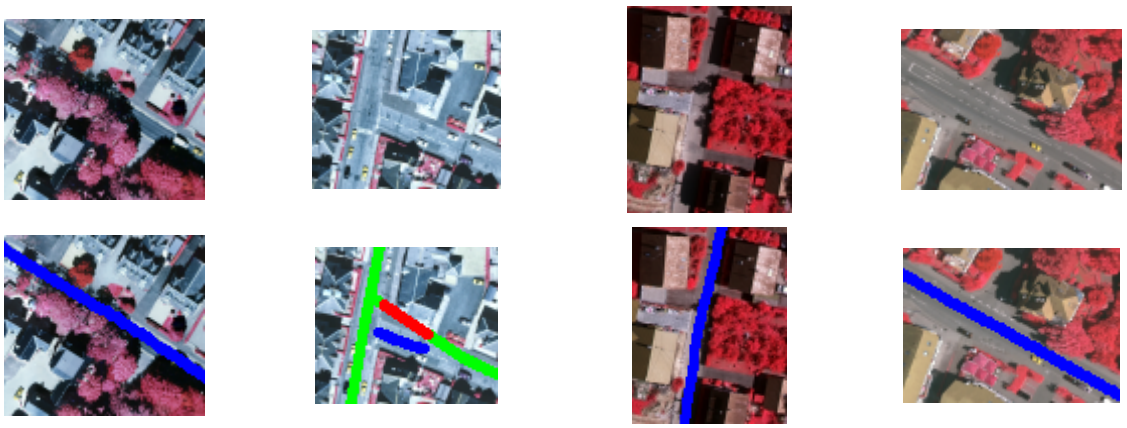
a. Grangemouth subset 1. *b. Grangemouth subset 2.*

c. Vaihingen subset 1. *d. Vaihingen subset 2.*

Figure 5.13. Results of road network extraction. Correctly extracted roads in green, incorrectly extracted roads in red, missed extractions in blue.

The extraction results show that the algorithm could extract the majority of the roads in the subsets. The correctness of the road extraction is quite high (better than 87 % for all examples), the completeness is lower. While the completeness in the Grangemouth examples is better than 70 %, it drops to 61 % in the first Vaihingen subset and to only 49 % in the second Vaihingen subset. One reason for this is that most of the parameters listed in Section 5.2 were set empirically using results of the Grangemouth data set, tuned such that they yield a high correctness. The images from the Vaihingen data set were processed with the same parameter values; the only parameter which was changed was the NDVI threshold. This indicates that at least some of the parameters are sensitive to sensor and scene characteristics. However, many roads could still be extracted, and the correctness remains high. The images show some situations in which the algorithm encounters difficulties. There are several places where trees occlude the road. Although the algorithm can deal with one tree, several trees in a row prevent road parts from being extracted (Figure 5.14a), and if the gap becomes too wide, it cannot be bridged. This is for example the case in the lower part of the Grangemouth subset 1 and in the upper right part of the Vaihingen subset 1. Geometric inaccuracies, especially towards the ends of roads, lead to both missed extractions and false extractions (Figure 5.14b). The reason for these geometric inaccuracies is that the correct end points of the road centre lines can be difficult to determine from the road parts when their borders have irregular shapes. The inaccuracies towards the ends of roads also affect the RMS, which lies between 1.42 m and 1.65 m. Another cause for missed extractions are roads whose width does not comply with the model, i.e. very narrow or very wide roads. In both Vaihingen subsets there are several rather narrow roads. For example in the Vaihingen subset 1 several narrow roads can be found in the upper right part (Figure 5.14c). In the upper right part of the Vaihingen subset 2 there is a rather wide road with several adjacent asphalt areas (Figure 5.14d). As these areas were merged with the road area, the resulting regions were too wide to be extracted as roads.

Mayer et al. (2006) claim that a completeness of at least 60 % and a correctness of at least 75 % is needed as a minimum for road extraction results to be considered practically useful; for real practical importance the completeness should be at least 70 % and the correctness at least 85 %. These goals are achieved for the Grangemouth subsets. For the Vaihingen subsets they are achieved for the correctness; however, for the completeness only the subset 1 fulfils at least the less stringent condition.



a. Road occluded by trees. *b. Geometric inaccuracy at end of road.* *c. Very narrow road.* *d. Very wide road.*

Figure 5.14. Causes for false and/or missed extractions. Correctly extracted roads in green, incorrectly extracted roads in red, missed extractions in blue.

Image subset	Completeness	Correctness	RMS
Grangemouth 1	77.6%	87.6%	1.42 m
Grangemouth 2	71.6%	94.5%	1.60 m
Vaihingen 1	61.5%	95.2%	1.58 m
Vaihingen 2	49.1%	91.9%	1.65 m

Table 5.6. Quantitative analysis of road extraction.

The road network which is formed by the extracted roads is topologically correct in all subsets, which means that all points which are connected in the extracted road network are also connected in the reference network. The topological completeness is lower; especially in the Grangemouth subset 2 the network is quite fragmented. This significant drop in the topological completeness is caused by several gaps which separate the connected components of the network. The gaps occur mainly between roads that should have been connected during the search for junctions. There are two main reasons for this failure of connection. One reason is a too large distance between the roads that should be connected (Figure 5.15a), especially if the search radius is small because the quality measures of the roads are low. Another reason is the geometric inaccuracy of the ends of roads (Figure 5.15b), which can lead to extensions going over vegetated areas or buildings and being rejected by the junction verification. This points to the necessity of a more sophisticated junction model and junction extraction method. The mean detour factor is relatively low for most subsets except for the Vaihingen subset 1, where it is higher mainly because of one missed junction connection in the centre of the image.

a. Missed connection due to small search radius. *b. Missed connection due to geometric inaccuracy.*

Figure 5.15. Missed junction connections. Correctly extracted roads in green, incorrectly extracted roads in red, missed extractions in blue.

Image subset	Mean detour factor	Top. completeness	Top. correctness
Grangemouth 1	1.05	75.5%	100%
Grangemouth 2	1.11	18.3%	100%
Vaihingen 1	1.27	59.6%	100%
Vaihingen 2	1.01	32.5%	100%

Table 5.7. Quantitative analysis of network topology.

Image subset	Junction completeness	Junction correctness	Junction RMS
Grangemouth 1	72.7%	84.2%	9.78 m
Grangemouth 2	59.3%	88.9%	3.59 m
Vaihingen 1	63.6%	70.0%	5.89 m
Vaihingen 2	30.8%	100%	8.88 m

Table 5.8. Quantitative analysis of junction extraction.

The junction extraction results are generally not as good as the results of the extraction of roads. As with the other results, the correctness of the junction extraction, being between 84% and 100%, is higher than the completeness, which is between 30% and 73%. Wrongly extracted junctions were most frequently caused by the extraction of parking lots adjacent to the roads. As stated above, missed junctions are often at places where a gap to a t-junction could not be bridged because the distance between both roads was too large, or because the verification of the junctions failed when the connection touched buildings or vegetated areas. Again, this points to the necessity of improving the junction extraction method. The RMS of the correctly extracted junctions lies between 3.5 m and 10 m. This rather low geometric accuracy of matched junctions is mainly caused by the lacking geometric accuracy of the road centre lines towards the ends of the roads.

5.3.3 Tests of Features for Road Extraction

The road extraction algorithm uses several different features in each step. In order to examine the impact of the features on the extraction, test runs were performed with six features for the images Grangemouth subset 1 and Vaihingen subset 1. In each test run one feature was not considered for the extraction, and the quality measures for the network were calculated. Some of the features are used in only one step, while others are used in several steps. If features are used in several steps, they are left out in all affected steps in the road extraction. Table 5.9 shows an overview on the tested features and in which steps they are used.

Criterion	Segmentation	Grouping	Road parts	Subgraphs	Network
NDVI	x	x	x		
Hue	x				
Height		x	x		
Border length		x			
Width constancy			x		
Context				x	x

Table 5.9. Overview on tested features. Steps where the respective feature is used are marked with x.

The results of the quantitative analysis of the test runs for the road extraction are shown in Table 5.10, the results of the topological analysis are shown in Table 5.11, and the results of the analysis of the junction extraction are shown in Table 5.12. The quality measures for the road extraction are the most significant ones, which is why the discussion of the results mainly focuses on them. Among the topological quality measures the topological completeness is the most significant measure, although not independent of the completeness of the extraction. The quality measures of the junction extraction also in general follow the trend of the results for the road extraction; the more dramatic changes in numbers are due to the low number of junctions: even one missed junction has an impact on the quality measure.

For most features, the extraction results decrease in quality if the respective feature is disregarded. This trend shows most clearly in the features NDVI and hue, which are used in the segmentation, the first step. For almost all quality measures, the values decrease dramatically if one of these features is not employed. For example, in the Grangemouth subset the completeness drops from nearly 78 % to barely over 50 % if the hue is not used. It is to be expected that the NDVI, which is used in three steps, influences the result to a great extent. The influence of the hue is not as self-evident, but it indicates the importance of the segmentation as the first step.

Disregarding the height feature corresponds to not using a DSM for the road extraction. As the algorithm was designed to be able to work with or without a DSM, the quality of the extraction does not decrease as drastically as for example without the NDVI, except for the topological correctness value in the Grangemouth subset. The topological correctness is usually 100%, but decreases here to below 90% because of some false connections via falsely extracted road parts. It is clear that the use of a DSM significantly increases the quality of the road extraction.

The influence of the context on the extraction is nearly negligible in the examined scenes because here the context objects mostly did not lead to a different decision in the subgraph evaluation than with interrelations alone. There are almost no branches, and context objects do not decide the validity of the connections in the subgraphs of these subsets. An exception is the topological correctness value in the Grangemouth subset, which drops from 100% to 97% because of one connection through a vegetated area. In order to show that the inclusion of context objects is nevertheless advisable, road extraction without using context objects was performed on the Grangemouth subset 2, where some roads are wrongly connected across buildings as a consequence. (Figure 5.16). The results of the quantitative analysis of the subset are recorded in Tables 5.10 to 5.12. The quality of the extraction is reduced in almost all quality measures; the correctness, for example, drops from 94.5% to 89.3%. The only quality measures that improve are the completeness and the topological completeness. However, the significant improvement of the topological completeness is caused by the wrong connections.

The influence of the other two features is more ambiguous. Both are used only in one step: the relative border length in the grouping, and the width constancy in the road part extraction. Where the quality measures decrease, they do not decrease as much as for the other features, and in some cases they increase, but not consistently: for example, while the completeness in the Grangemouth subset drops if the border length is disregarded, it increases for the Vaihingen subset.

Feature	Subset	Completeness		Correctness		RMS	
NDVI	Gr	58.8%	(-18.8%)	67.0%	(-20.6%)	1.61 m	(-0.19 m)
	Va	37.9%	(-23.6%)	80.4%	(-14.8%)	1.83 m	(-0.25 m)
Hue	Gr	50.3%	(-27.3%)	77.6%	(-10.0%)	1.40 m	(+0.02 m)
	Va	45.8%	(-15.7%)	88.0%	(-7.2%)	1.57 m	(+0.01 m)
Height	Gr	72.5 %	(-5.1%)	83.3%	(-4.3%)	1.61 m	(-0.19 m)
	Va	46.5%	(-15%)	86.5%	(-8.7%)	1.47 m	(+0.11 m)
Border length	Gr	70.1%	(-7.5%)	89.8%	(+2.2%)	1.24 m	(+0.18 m)
	Va	71.4%	(+9.9%)	92.0%	(-3.2%)	1.74 m	(-0.16 m)
Width const.	Gr	71.9%	(-5.7%)	82.5%	(-5.1%)	1.50 m	(-0.08 m)
	Va	66.4%	(-11.2%)	92.2%	(+4.6%)	1.51 m	(-0.09 m)
Context	Gr	79.8%	(+2.2%)	87.4%	(-0.2%)	1.47 m	(-0.05 m)
	Va	61.7%	(+0.2%)	95.4%	(+0.2%)	1.59 m	(-0.01 m)
	Gr 2	74.1%	(+2.5%)	89.3%	(-5.2%)	1.66 m	(-0.06 m)

Table 5.10. Impact of features on road extraction. Gr = Grangemouth subset 1, Va = Vaihingen subset 1, Gr 2 = Grangemouth subset 2; numbers in brackets give difference to original results shown in Table 5.6, negative numbers indicate worse result.

Figure 5.16. False extractions in Grangemouth subset 2 if context objects are not used. Correctly extracted roads in green, incorrectly extracted roads in red, missed extractions in blue.

Feature	Subset	Mean detour factor		Top. completeness		Top. correctness	
NDVI	Gr	1.13	(-0.08)	28.7%	(-46.8%)	100%	(0)
	Va	1	(+0.27)	17.3%	(-42.3%)	100%	(0)
Hue	Gr	1.06	(-0.01)	29.9%	(-45.5%)	100%	(0)
	Va	1	(+0.27)	17.3%	(-42.3%)	99.7%	(-0.3%)
Height	Gr	1.15	(-0.1)	67.4%	(-8.1%)	89.6%	(-10.4%)
	Va	1	(+0.27)	31.0%	(-28.6%)	100%	(0)
Border length	Gr	1.01	(+0.26)	42.0%	(-33.5%)	100%	(0)
	Va	1.49	(-0.22)	100%	(+40.4%)	100%	(0)
Width const.	Gr	1.12	(-0.07)	71.8%	(-3.7%)	100%	(0)
	Va	1.54	(-0.27)	96.1%	(+36.5%)	100%	(0)
Context	Gr	1.08	(-0.03)	75.5%	(0)	96.9%	(-3.1%)
	Va	1.27	(0)	59.6%	(0)	100%	(0)
	Gr 2	1.54	(-0.43)	45.2%	(+26.9%)	97.3%	(-2.7%)

Table 5.11. Impact of features on network topology. Gr = Grangemouth subset 1, Va = Vaihingen subset 1; numbers in brackets give difference to original results shown in Table 5.7, negative numbers indicate worse result.

Feature	Subset	Junction completeness		Junction correctness		Junction RMS	
NDVI	Gr	36.4%	(-36.3%)	34.8%	(-49.4%)	6.83 m	(+2.95 m)
	Va	20.0%	(-43.6%)	50.0%	(-20.0%)	9.87 m	(-3.98 m)
Hue	Gr	36.4%	(-36.3%)	53.3%	(-30.9%)	6.75 m	(+3.03 m)
	Va	13.3%	(-50.3%)	33.3%	(-36.7%)	9.00 m	(-3.11 m)
Height	Gr	72.7%	(0)	72.7%	(11.5 %)	10.00 m	(-0.22 m)
	Va	13.3%	(-50.3%)	40.0%	(-30.0%)	3.90 m	(-1.99 m)
Border length	Gr	54.5%	(-18.2%)	66.7%	(-17.5%)	3.27 m	(+6.51 m)
	Va	46.7%	(-16.9%)	70.0%	(0)	4.84 m	(-1.05 m)
Width const.	Gr	54.5%	(-18.2%)	57.1%	(-27.1%)	5.32 m	(+4.46 m)
	Va	46.7%	(-16.9%)	70.0%	(0)	3.03 m	(+2.86 m)
Context	Gr	63.6%	(-9.1%)	70.0%	(-14.2%)	7.02 m	(+2.76 m)
	Va	40.0%	(-23.6%)	66.7%	(-3.3%)	5.73 m	(+0.16 m)
	Gr 2	59.3%	(0)	76.2%	(-12.7%)	4.24 m	(-0.65 m)

Table 5.12. Impact of features on junction extraction. Gr = Grangemouth subset 1, Va = Vaihingen subset 1; numbers in brackets give difference to original results shown in Table 5.8, negative numbers indicate worse result.

5.3.4 Comparison with Other Approaches

In order to compare the results of the approach with those of other approaches, Table 5.13 shows an overview of some other approaches for road extraction in urban areas together with the average of the results for all four examined subsets. Of course, since different scenes were used for tests of the different approaches, the comparison is not ideal. The GSD of the images used in the compared approaches lies between 0.08 m and 1 m. The buffer width for the calculation of the quality measures is ±5 m for all approaches where it was stated. The best results for completeness (80 %) are achieved in (Youn et al., 2008) and (Poullis and You, 2010); however, they do not give a buffer width, which makes the comparison less accurate. It also has to be noted that in (Poullis and You, 2010), the network is completed interactively; the authors do not state whether the quality measures are computed for the results before or after the completion. The lowest completeness (49.5 %) is reported of the approach in (Zhang and Couloigner, 2006). The completeness of the approach presented in this thesis is significantly higher, but still lower than that of most of the other approaches. The correctness, on the other hand, is higher than that of most of the other approaches, with the exception of (Hinz, 2004), where a slightly higher correctness is reported. However, Hinz (2004) only gives results for two small test sites that are, in addition, from the same scene. Results concerning the transferability of the approach are not reported. The RMS of the approach is also good in comparison with the other approaches; it is the second lowest of all approaches where a RMS was reported, and the lowest of all approaches within the same resolution range of 10 to 20 cm. The topological completeness and correctness is only reported in (Hinz, 2004), where the topological completeness is significantly higher than in the approach of this thesis. However, the test sites used in (Hinz, 2004) have three to four junctions as opposed to, for instance, 20 junctions in the Grangemouth data set of our approach.

	Hinz, 2004	Doucette et al., 2004	Zhang and Couloigner, 2006	Youn et al., 2008	Poullis and You, 2010	Approach in this thesis
Completeness	79.1%	65.0%	49.5%	80.0%	80.6%	65.0%
Correctness	96.9%	83.0%	43.0%	79.0%	75.3%	92.3%
RMS	1.9 m	6.63 m	1.52 m	2.32 m	-	1.56 m
top. completeness	92.0 %	-	-	-	-	46.4%
top. correctness	98.1 %	-	-	-	-	100%
GSD	< 0.2 m	1.0 m	1.0 m	0.1 m	?	0.08-0.01m
Buffer width	±5 m	?	±5 m	?	?	±5 m

Table 5.13. Comparison of quality measures. If several results were given in one source, the average was calculated; this applies to the approach described in this thesis as well.

6 Conclusions and Outlook

6.1 Conclusions

In this thesis, a method for the automatic extraction of a road network from aerial images was presented, for the specific environment of suburban areas. The approach consists of several successive steps; it is region-based, starting with a segmentation of the image. In each step, several radiometric and geometric criteria are combined using specific knowledge about the appearance of roads in suburban areas. A DSM is used as additional source of information, but the algorithm can also be employed without a DSM. The result is a road network.

The road extraction starts with an initial segmentation of the image using the normalized cuts algorithm with mainly radiometric criteria. The aim of the segmentation is to have as many road borders as possible coincide with segment borders. The oversegmentation which is the outcome of the initial segmentation is compensated for in a subsequent grouping step using both radiometric and geometric criteria. After the grouping, road parts are extracted, again using radiometric and geometric criteria. In this step, roads are not required to be extracted in one part from junction to junction; one road can consist of several road parts. This allows for the existence of disturbances in the appearance of the road in the image, which can be caused by context objects or changing road surface materials. Gaps between road parts are bridged in the next step, where road subgraphs are generated by connecting road parts that are likely to belong to the same road according to geometric criteria. In this process, the presence of false extractions can lead to branches in the subgraphs, which represent different possible courses of the road. Therefore, the connections between road parts in the subgraphs are evaluated in order to eliminate those connections that are most likely to be false. The connections are evaluated using two types of information: the geometric and radiometric interrelations between connected road parts, and the presence of context objects in the gap between the road parts. Context objects used in the approach are buildings, trees, vehicles, vegetated areas and road surface areas. After the subgraph evaluation, the subgraphs consist of single road strings without branches. Approximated centre lines are derived from these road strings, and the network is generated using these centre lines. The network is built by searching for junctions at the ends of the centre lines. The final road network consists of the centre lines of the roads, with associated widths, and the junction points. One road centre line stretches from junction to junction.

The results show that the algorithm can extract roads in suburban areas with good results. The minimum completeness and correctness conditions given in (Mayer et al., 2006) for a practically relevant extraction approach (see Chapter 5.3.2) are met for the Grangemouth subsets; for the Vaihingen subsets, they are only met for the correctness condition, and for the lower completeness condition in one subset. The correctness of the extraction is generally about 90 %. False extractions mainly occur in parking lots next to the roads. Another cause for extractions counted as false are geometric inaccuracies at the ends of roads, where extracted roads do not lie in the buffer around the reference road. The completeness is not as good as the correctness. More than 70 % completeness was achieved in both Grangemouth scenes, but the completeness drops to 61 % in the Vaihingen subset 1 and 49 % in the Vaihingen subset 2. The lack of completeness is often caused by trees or building shadows which cover large parts of some roads, especially in the Vaihingen data sets. Another cause for lacking completeness are roads that are very wide or very narrow, such that no road parts could be extracted that fit the geometric criteria. Part of the drop in completeness in the Vaihingen subsets can be ascribed to the fact that the parameter settings were empirically determined from test subsets in the Grangemouth scene and transferred to the Vaihingen

scenes without change, except for the parameter value of the NDVI threshold. The correctness of the Vaihingen subsets is still high, and at least half of the roads could be extracted, which demonstrates that the method can be used on different scenes. In general, the Vaihingen scenes are more difficult for extraction because there are more parts of the roads occluded by trees, and the widths of the roads vary more.

Generally, in order to extract a road, the road model requires at least parts of the road to be relatively homogeneous and distinct from the surroundings, by edges or by colour. These parts need to meet the minimum length and maximum width criteria for the road parts to be successfully extracted. This means that the extraction fails if the road surface is so fragmented that no reliable road parts could be found.

The quality of the network topology is generally correlated with the completeness of the network, but not always: although the completeness of the extraction in the Grangemouth subset 2 is above 70%, the topological completeness is very low. This significant drop in the topological completeness is caused by several gaps which separate the connected components of the network. The gaps between the connected components were often not bridged because of inaccurate location of the road ends, leading to a rejection of the connection in the junction search because of interfering context objects. Consequently, the last step of the approach, the network generation, should be improved for practical applications.

The comparison with other approaches (Chapter 5.3.2; Table 5.9) shows that the completeness and correctness values of the approach presented here are within the range of values reported for other road extraction approaches in urban and suburban areas. In terms of correctness, it is better than most other approaches. The only approach that is better on both accounts is that in (Hinz, 2004). However, in that approach, only two small image subsets from the same scene were used to compute the results. In addition, the approach by (Hinz, 2004) uses road markings extensively; it is not shown how well it would work in an area without road markings. The approach by (Hinz, 2004) also requires a DSM, whereas the approach presented here can also be used when no DSM is available. The comparison of the approaches also shows that using high resolution images has benefits for road extraction in urban areas compared to using low resolution images, as the overall best results were achieved with the approaches using high resolution images in the 10-20 cm range.

The results of the tests without some features show the importance of these features. The comparison between the results with and without the DSM, for example, shows that the use of the DSM improves the extraction considerably. Although the method can still produce quite reasonable results without a DSM, the DSM should be used if it is available. Another important feature is the NDVI, which is used in several steps of the approach. If the NDVI is not used, this leads to a significant drop in the quality of the extraction. In general, features used in early steps have larger influence on the final result, which shows the importance of the early steps. A segmentation which incorporates knowledge about roads and different features, as used here with the normalized cuts method, provides a solid foundation for the later steps.

If the road part extraction is reliable, which it usually is, especially when a DSM is used, the subgraph evaluation step is often of lesser importance. False road part extractions are relatively rare and mostly isolated enough from other road parts that branches in the subgraphs are seldom formed. Still, even in this case the evaluation using the context objects is helpful for the elimination of wrong connections between road parts that cross buildings or vegetated areas.

6.2 Outlook

There are several areas where the extraction quality could be improved. The most important ones are the completeness of the extraction, especially in the network generation phase and in the presence of disturbances, and the geometric quality, especially of the junctions.

The completeness of the extraction can be improved by improving the search for junctions. In the current implementation, a very simple junction model is used. The use of a more sophisticated junction model and a separate extraction of junctions, where image information in the junction area is used more extensively, could improve the topological completeness in particular. To achieve this, junction locations could be hypothesised at the ends of roads and checked by a separate junction extraction and evaluation. This approach of a more sophisticated junction evaluation would allow the search radius for junctions to be larger and consequently more junctions could be extracted.

Another way of improving the completeness could be an additional step of closing further gaps after the final network check. One possibility is to search for reasonable connections between the unconnected ends of roads and junction points, possibly using road parts that are extracted with more relaxed parameter settings. However, this must be done with care to prevent the connection of dead ends to other roads. A similar search could be done with the isolated roads that are currently deleted during the final network check. One could also use an improved modelling of the context objects in order to bridge larger gaps in the network generation, for example rows of trees or rows of building shadows. An improved modelling of context objects could also help to improve the performance in earlier steps. Rows of trees or rows of building shadows in conjunction with road surface areas could be used to extract a second set of road parts that are less reliable but could be used to expand road subgraphs.

The geometric accuracy of the final network could be improved using a network snake-based approach. Another possibility is to use the extracted road centre lines as approximate locations of the true centre lines that help to precisely detect the edges delineating the road. For the improvement of the geometric accuracy of the junctions, a more sophisticated junction modelling and extraction method as described above would be helpful in addition to improving the completeness.

Another area which could be improved is the setting of parameters. The algorithm uses a relatively large number of parameters which are set empirically using the Grangemouth data set. When the algorithm is applied to the Vaihingen data set, the completeness decreases significantly. In the current implementation, only one parameter (the NDVI threshold) is changed when the data sets are switched. The value of this parameter must be determined by the user. It is conceivable that the results can be improved if the parameters are adapted better to the scene. Therefore, it should be determined which other parameters have to be changed between different scenes to achieve a better completeness. Also, a systematic training of the parameters using a stochastic model would be a better way to set the parameters, in order to enhance the robustness of the algorithm.

Appendix – List of Rules for Merging

Five features are combined to determine whether two segments can be merged or not: the absolute length of the shared border (ABL), the relative length of the shared border (RBL), the convexity (C), the edge strength (ES) and the histogram difference (HD). For each feature, the fuzzy membership values are computed for either two or three sets. The two sets *ok_for_merging* (*ok*) and *not_ok_for_merging* (*nok*) are used for all five features; for the features ES and HD, the set *limited_ok_for_merging* (*lok*) is used additionally. A region pair is said to *belong* to a set with respect to a feature if its membership value for this set is larger than for the other set(s). If the membership value is 1, the pair is considered to *belong fully* to the respective set. Using the membership values of the sets, a set of rules is employed in order to decide whether the pair of segments may be merged or not:

if region pair *belongs* to set *ok* for all features
    if region pair *belongs fully* to set *ok* for at least one feature
    or shared border lies on division border
        then region pair can be merged
else if region pair *belongs* to set *ok* for all features except for RBL
    if region pair *belongs fully* to set *ok* for at least two features
    or shared border lies on division border
        then region pair can be merged
else if region pair *belongs* to set *ok* for merging for the features ABL, RBL and C
    if region pair *belongs* to set *lok* for ES
        if region pair *belongs fully* to set *ok* for HD
            if region pair *belongs fully* to set *ok* for at least one other feature
            or shared border lies on division border
                then region pair can be merged
    else if region pair *belongs* to set *lok* for HD
        if region pair *belongs fully* to set *ok* for ES
            if region pair *belongs fully* to set *ok* for at least one other feature
            or shared border lies on division border
                then region pair can be merged
    else if region pair *belongs* to set *nok* for ES
        if ES value for *nok* – ES value for *lok* < 0.2
            if region pair *belongs fully* to set *ok* for at least two features
            or (shared border lies on division border and region pair *belongs fully* to set *ok* for at least one feature)
                then region pair can be merged
    else if region pair *belongs* to set *nok* for HD
        if HD value for *nok* – HD value for *lok* < 0.2
            if region pair *belongs fully* to set *ok* for at least two features
            or (shared border lies on division border and region pair *belongs fully* to set *ok* for at least one feature)
                then region pair can be merged
    else if region pair *belongs* to set *lok* for ES and HD
        if region pair *belongs fully* to set *ok* for at least two features
        or (shared border lies on division border and region pair *belongs fully* to set *ok* for at least one feature)
            then region pair can be merged
else region pair cannot be merged

References

Azran, A. and Ghahramani, Z. (2006) Spectral methods for automatic multiscale data clustering. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (New York), 17-22 June, Vol. 1, pp. 190-197.

Bacher, U. and Mayer, H. (2005) Automatic road extraction from multispectral high resolution satellite images. Proc. CMRT05 (Vienna), Aug 29-30, IAPRSSIS Vol. XXXVI, Part 3/W24, pp. 29-34.

Bajcsy, R. and Tavakoli, M. (1976) Computer recognition of roads from satellite pictures. *IEEE Transactions on Systems, Man, and Cybernetics* **6**(9), 623-637.

Barsi, A., Heipke, C. and Willrich, F. (2002) Junction extraction by artificial neural network system – JEANS. Proc. PCV02 (Graz), Sep 9-13, IAPRSSIS Vol. XXXIV. Part 3B, pp. 18-21.

Baumgartner, A., Steger, C., Mayer, H., Eckstein, W. and Ebner, H. (1999) Automatic road extraction based on multi-scale, grouping, and context. *Photogrammetric Engineering & Remote Sensing* **65**(7), 777-785.

Baumgartner, A. (2003) Automatische Extraktion von Straßen aus digitalen Luftbildern. Dissertation. Deutsche Geodätische Kommission, Reihe C, Vol. 564 (in German).

Beucher, S. and Meyer, F. (1993) 'The morphological approach to segmentation: the watershed transformation'. In: *Mathematical Morphology in Image Processing*. Ed: E. R. Dougherty. Marcel Dekker, New York. pp. 433-481.

Boykov, Y., Veksler, O. and Zabih, R. (2001) Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(11), 1222-1239.

Burger, W. and Burge, M.J. (2006) *Digitale Bildverarbeitung: Eine Einführung mit Java und ImageJ*. Second Edition. Springer-Verlag Berlin Heidelberg (in German).

Clode, S., Rottensteiner, F., Kootsookos, P. and Zelniker, E. (2007) Detection and vectorization of roads from Lidar data. *Photogrammetric Engineering & Remote Sensing* **73**(5), 517-536.

Comaniciu, D. and Meer, P. (2002) Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(5), pp. 603-619.

Cramer, M. (2010) The DGPF-Test on digital airborne camera evaluation – overview and test design. *PFG* **2**(2010): 69-72.

Dantzig, G.B. (1951) 'Maximization of a linear function of variables subject to linear inequalities'. In: *Activity Analysis of Production and Allocation*. Ed: T.C. Koopmans. Jon Wiley & Sons, New York. pp. 339-347.

Dantzig, G.B. and Thapa, M.N. (1997) *Linear Programming 1: Introduction*. Springer Series in Operations Research. Springer-Verlag New York.

Doucette, P., Agouris, P., Stefanidis, A. and Musavi, M. (2001) Self-organised clustering for road extraction in classified imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* **55**(5-6), 347-358.

Doucette, P., Agouris, P. and Stefanidis, A. (2004) Automated road extraction from high resolution multispectral imagery. *Photogrammetric Engineering & Remote Sensing* **70**(12), 1405-1416.

Dal Poz, A.P., Zanin, R.B. and do Vale, G.M. (2006) Automated extraction of road network from medium- and high-resolution images. *Pattern Recognition and Image Analysis*, **16**(2), 239-248.

Eckstein, W. (1996) Segmentation and texture analysis. Proc. XVIII. ISPRS Congress (Vienna), Jul 12-18, IAPRSSIS XXXI, Part B3, pp. 165-175.

ERF (2010) European road statistics 2010. European Union Road Federation, Brussels.

Estrada, F.J. and Jepson, A.D. (2004) Spectral embedding and min-cut for image segmentation. In: British Machine Vision Conference, pp. 317-326.

Estrada, F.J. and Jepson, A.D. (2009) Benchmarking image segmentation algorithms. *International Journal of Computer Vision* **85**(2), pp. 167-181.

Felzenszwalb, P.F. and Huttenlocher, D.P. (1998) Image segmentation using local variation. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 98-104.

Fischler, M.A., Tenenbaum, J.M. and Wolf, H.C. (1981) Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique. *Computer Graphics and Image Processing* **15**(3), 201-223.

Frizzelle, B.G., Evenson, K.R., Rodriguez, D.A. and Laraia, B.A. (2009) The importance of accurate road data for spatial applications in public health: customizing a road network. *International Journal of Health Geographics* **8**(24) (May 2009).

Fua, P. (1996) Model-based optimization: accurate and consistent site modeling. Proc. XVIII. ISPRS Congress (Vienna), Jul 12-18, IAPRSSIS XXXI Part B3/III, pp. 222-233.

Gabet, L., Giraudon, G. and Renouard, L. (1997) Automatic generation of high resolution urban zone digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing* **52**(1), 33-47.

Géraud, T. and Mouret, J.-B. (2004) Fast road network extraction in satellite images using mathematical morphology and Markov random fields. *EURASIP Journal on Applied Signal Processing* **2004**(16), 2503-2514.

Gerke, M. (2006) Automatic quality assessment of road databases using remotely sensed imagery. Dissertation. Deutsche Geodätische Kommission, Reihe C, Vol. 599, 103 p.

Gerke, M. and Heipke, C. (2008) Image-based quality assessment of road databases. *International Journal of Geographical Information Science* **22**(8), 871-894.

Gibson, L. (2003) Finding road networks in Ikonos satellite imagery. Proc. Annual ASPRS Conference (Anchorage, AK), May 5-9, on CD-ROM.

Göpfert, J. and Rottensteiner, F. (2009) Adaptation of roads to ALS data by means of network snakes. Proc. Laserscanning09 (Paris), Sep 1-2, IAPRSSIS XXXVIII Part 3/W8, pp. 24-29.

Golub, G.H. and Van Loan, C.F. (1996) *Matrix Computations*. Third Edition. Johns Hopkins University Press, Baltimore.

Green, P. (1995) Reversible jump MCMC computation and Bayesian model determination. *Biometrika* **82**(4): 711-732.

Grün, A. and Li, H. (1995) Road extraction from aerial and satellite images by dynamic programming. *ISPRS Journal of Photogrammetry and Remote Sensing* **50**(4), 11-20.

Haverkamp, D. (2002) Extracting straight road structure in urban environments using IKONOS satellite imagery. *Optical Engineering* **41**(9), 2107-2110.

Hinz, S. (2004). Automatische Extraktion urbaner Straßennetze aus Luftbildern. Dissertation. Deutsche Geodätische Kommission, Reihe C, Vol. 580 (in German), 142 p.

Hinz, S. and Baumgartner, A. (2003). Automatic extraction of urban road networks from multi-view aerial imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* **58**(1-2), 83-98.

Hu, J., Razdan, A., Feminani, J.C., Cui, M. and Wonka, P. (2007) Road network extraction and intersection detection from aerial images by tracking road footprints. *IEEE Transactions on Geoscience and Remote Sensing* **45**(12), 4144-4157.

Hu X., Zhang, Z. and Tao, C.V. (2004a) A robust method for semi-automatic extraction of road centerlines using a piecewise parabolic model and least square template matching. *Photogrammetric Engineering & Remote Sensing* **70**(12), 1393-1398.

Hu X., Tao, C.V. and Hu Y. (2004b) Automatic road extraction from dense urban area by integrated processing of high resolution imagery and LIDAR data. Proc. XX. ISPRS Congress (Istanbul), Jul 12-23, IAPRSSIS XXXV Part B3, pp. 288-292.

Hu, X. and Tao, C.V. (2007) Automatic extraction of main road centerlines from high resolution satellite imagery using hierarchical grouping. *Photogrammetric Engineering & Remote Sensing* **73**(9), 1049-1056.

Jain, A.K. and Dubes, R.C. (1988) *Algorithms for Clustering Data*. Prentice Hall, Inc., Englewood Cliffs, New Jersey.

Kass, M., Witkin, A. and Terzopoulos, D. (1987) Snakes: Active Contour Models. *International Journal of Computer Vision* **1**(4): 321-331.

Klang, D. (1998) Automatic detection of changes in road databases using satellite imagery. Proc. ISPRS Commission IV Symposium on GIS – Between Visions and Applications (Stuttgart), IAPRS 32(4), pp. 293-298.

Klir, J. (2006) *Uncertainty and Information: Foundations of Generalized Information Theory*. John Wiley & Sons, Hoboken, NJ.

van Lieshout, M.N.M. (2000) *Markov Point Processes and their Applications*. World Scientific Publishing Company, Singapore.

Lin, X. G., Zhang, J.X., Liu, Z.J. and Shen, J. (2008) Semi-automatic extraction of ribbon roads from high resolution remotely sensed imagery by cooperation between angular texture signature and template matching. Proc. XXI. ISPRS Congress (Beijing), Jul 3-11, IAPRSSIS XXXVII Part B3b, pp. 539-544.

Lucchese, L. and Mitra, S.K. (2001) Color image segmentation: a state-of-the-art survey. Image Processing, Vision and Pattern Recognition, Proc. of the Indian National Science Academy (INSA-A), 67 A(2), 207-221.

Luo, J. and Guo, C. (2003) Perceptual grouping of segmented regions in color images. *Pattern Recognition* **36**(12), 2781-2792.

Ma L., Li, J. and Chen, J. (2008) Upgrading fundamental GIS databases for navigation from high resolution satellite imagery. Proc. XXI. ISPRS Congress (Beijing), Jul 3-11, IAPRSSIS XXXVII Part B6b, pp. 335-338.

Malladi, R., Sethian, J.A. and Vemuri, B.C. (1995) Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17**(2), 158-175.

Marr, D. and Hildreth, E. (1980) Theory of edge detection. *Proceedings of the Royal Society of London*, Series B, Biological Sciences, **207**(1167), 187-217.

Mayer, H., Hinz, S., Bacher, U. and Baltsavias, E. (2006) A test of automatic road extraction approaches. Proc. PCV06 (Bonn), Sep 20-22, IAPRSSIS Vol. XXXVI Part 3, pp. 209-214.

McKeown, D.D. and Denlinger, J.L. (1988) Cooperative methods for road tracking in aerial imagery. Proc. Computer Society Conference on Computer Vision and Pattern Recognition (Ann Arbor, MI), Jun 5-9, pp. 662-672.

Mena, J.B. and Malpica, J.A. (2005) An automatic method for road extraction in rural and semi-urban areas starting from high resolution satellite imagery. *Pattern Recognition Letters* **26**(2005), 1201-1220.

Meyer, H. (2009) Verwendung von Kontextobjekten für die Extraktion von Straßen in Vorstadtgebieten aus Luftbildern. Master's Thesis, Leibniz Universität Hannover (unpublished).

Mohammadzadeh, A., Tavakoli, A. and Valadan Zoej, M.J. (2006) Road extraction based on fuzzy logic and mathematical morphology from pan-sharpened IKONOS images. *The Photogrammetric Record* **21**(113), 44-60.

Nevatia, R. and Babu, K.R. (1980) Linear feature extraction and description. *Computer Graphics and Image Processing* **13**(3), 257-269.

Nocedal, J. and Wright, S.J. (2006) *Numerical Optimization*. Springer Series in Operations Research. Second Edition. Springer-Verlag New York.

Papadimitriou, C.H. and Steiglitz, K. (1998) *Combinatorial optimization: algorithms and complexity*. Dover Publications, Mineola, NY.

Peng, T., Jermyn, I.H., Prinet, V. and Zerubia, J. (2008) Extraction of main and secondary roads in VHR images using a higher-order phase field model. Proc. XXI. ISPRS Congress (Beijing), Jul 3-11, IAPRSSIS XXXVII Part B3a, pp. 215-222.

Perez, F. and Koch, C. (1994) Toward color image segmentation in analog VLSI: algorithm and hardware. *International Journal of Computer Vision* **12**(1), 17-42.

Poullis, C. and You, S. (2010) Delineation and geometric modelling of road networks. *ISPRS Journal of Photogrammetry and Remote Sensing* **65**(2), 165-181.

Price, K. (1999) Road grid extraction and verification. Proc. Workshop "Automatic Objects from Digital Imagery" (Munich), Sep 8-10, IAPRSSIS XXXII Part 3-2W5, pp. 101-106.

Quam, L.H. (1978) Road tracking and anomaly detection in aerial imagery. Technical Note 158, SRI International, Menlo Park, CA, 5 p.

Ramer, U. (1972) An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing* **1**(3), 244-256.

Ravanbakhsh, M., Heipke, C. and Pakzad, K. (2008a) Road junction extraction from high resolution aerial imagery. *The Photogrammetric Record* **23**(124), 405-423.

Ravanbakhsh, M., Heipke, C. and Pakzad, K. (2008b)Automatic extraction of traffic islands from aerial images. *PFG* **5**(2008), 375-384.

Ravanbakhsh, M. and Fraser, C. (2009) Road roundabout extraction from very high resolution aerial imagery. Proc. CMRT09 (Paris), Sep 3-4, IAPRSSIS XXXVIII Part 3/W4, pp. 19-26.

Rochery, M., Jermyn, I. and Zerubia, J. (2005) Phase field models and higher-order active contours. Proc. IEEE International Conference on Computer Vision (Beijing), Oct 17-20, Vol. 2, pp. 970-976.

Rouse, J.W., Jr., R.H. Haas, J.A. Schell, and D.W. Deering (1973) Monitoring the vernal advancement and retrogradation (green wave effect) of natural vegetation. Prog. Rep. RSC 1978-1, Remote Sensing Center, Texas A&M Univ., College Station (NTIS No. E73- 106393).

Rubner, Y. (1999) Perceptual metrics for image database navigation. Ph.D. thesis, Stanford University.

Rubner, Y., Puzicha, J., Tomasi, C. and Buhmann, J.M. (2001) Empirical evaluation of dissimilarity measures for colour and texture. *Computer Vision and Image Understanding* **84**(1): 25-43.

Ruskoné, R. and Airault, S. (1997) 'Toward an automatic extraction of the road network by local interpretation of the scene'. In: *Photogrammetric Week '97*. Eds: D. Fritsch and D. Hobbie. Wichmann Verlag, Heidelberg, pp. 147-157.

Shackelford, A.K. and Davis, C.H. (2003) Fully automated road network extraction from high-resolution satellite multispectral imagery. Proc. International Geoscience and Remote Sensing Symposium (IGARSS) (Toulouse), Jul 21-25, Vol. 1, pp. 461-463.

Shawe-Taylor, J. and Cristianini, N. (2004) *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge.

Shi, J. and Malik, J. (2000) Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(8), 888-905.

Soille, P. (1999) *Morphological Image Analysis*. Springer Verlag Berlin.

Steger, C. (1998) An unbiased detector of curvilinear structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(2), 113-125.

Stilla, U. (1995) Map-aided structural analysis of aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing* **50**(4), 3-10.

Stoica, R., Descombes, X. and Zerubia, J. (2004) A Gibbs point process for road extraction from remotely sensed images. *International Journal of Computer Vision* **57**(2), 121-136.

Ton, J., Jain, A.K., Enslin, W.R. and Hudson, W.D. (1989) Automatic road identification and labeling in Landsat 4 TM images. *Photogrammetria* **43**(5), 257-276.

Vosselman, G. and de Knecht, J. (1995). 'Road tracing by profile matching and Kalman filtering'. In: *Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*. Eds: A. Gruen, O. Kuebler, P. Agouris. Birkhäuser Verlag, Basel. pp. 265-274.

Wang, F. and Newkirk, R. (1988) A knowledge-based system for highway network extraction. *IEEE Transactions on Geoscience and Remote Sensing* **26**(5), 525-531.

Weidner, U. and Förstner, W. (1995) Towards automatic building extraction from high-resolution digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing* 50(4), 38-49.

Wiedemann, C., Heipke, C., Mayer, H. and Jamet, O. (1998) 'Empirical evaluation of automatically extracted road axes'. In: *Empirical Evaluation Methods in Computer Vision*. Eds: K.J. Boyer and P.J. Phillips. IEEE Computer Society Press, Los Alamitos, CA, pp. 172-187.

Wiedemann, C. and Ebner, H. (2000) Automatic completion and evaluation of road networks. Proc XIX. ISPRS Congress (Amsterdam), Jul 16-22, IAPRSSIS XXXIII Part B3/2, pp. 979-986.

Wiedemann, C. (2002) Extraktion von Straßennetzen aus optischen Satellitenbilddaten. Dissertation. Deutsche Geodätische Kommission, Reihe C, Vol. 551, 94 p. (in German).

Wu, Z. and Leahy, R. (1993) An optimal graph theoretic approach to data clustering: theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**(11), 1101-1113.

Youn, J., Bethel, J.S., Mikhail, E.M. and Lee, C. (2008) Extracting urban road networks from high-resolution true orthoimage and lidar. *Photogrammetric Engineering & Remote Sensing* **74**(2), 227-237.

Yu, S.X. and Shi, J. (2003) Multiclass spectral clustering. Proc. 9th IEEE International Conference on Computer Vision (Nice), Oct 13-16, Vol. 2, pp. 313-319.

Zadeh, L.A. (1965) Fuzzy Sets. *Information and Control* **8**, 338-353.

Zhang, C. (2004) Towards an operational system for automated updating of road databases by integration of imagery and geodata. *ISPRS Journal of Photogrammetry and Remote Sensing*, **58**(3-4), pp. 160-168.

Zhang, Q. and Couloigner, I. (2006). Automated road network extraction from high resolution multi-spectral imagery. Proc. ASPRS Annual Conference (Reno, Nevada), May 1-5, on CD-ROM.

Zhou, J., Bischof, W.F. and Caelli, T. (2006) Road tracking in aerial images based on human-computer interaction and Bayesian filtering. *ISPRS Journal of Photogrammetry and Remote Sensing* **61**(2), 108-124.

Ziems, M., Gerke, M. and Heipke, C. (2007) Automatic road extraction from remote sensing imagery incorporating prior information and colour segmentation. In: Stilla, U. et al. (Eds.) PIA07. IAPRSSIS 36(3/W49A), pp. 141-147.

Ziems, M., Fujimura, H., Heipke, C. and Rottensteiner, F. (2010) Multiple-model based verification of Japanese road data. Proc. ISPRS Workshop "Core Spatial Databases – Updating, Maintenance and Services – from Theory to Practice" (Haifa), Mar 15-17, IAPRSSIS XXXVIII Part 4-8-2/W9, pp. 13-19.

Acknowledgement - Danksagung

Zuallererst gilt mein Dank meinem Doktorvater Prof. Dr.-Ing. Christian Heipke, der mir die Möglichkeit gegeben hat, diese Arbeit anzufertigen, und der mich immer wieder unterstützt hat. Er hat mir auch ermöglicht, die Arbeit nach dem Ende meines Projektes abzuschließen, und oft größeres Zutrauen zu meinen Fähigkeiten gezeigt als ich selbst.

Mein Dank gilt auch den anderen Referenten: Dr.-Ing. Markus Gerke vom ITC in Enschede, der außerdem durch seine Vorarbeiten die Voraussetzungen für meine Stelle geschaffen hat, Prof. Dr.-Ing. Monika Sester vom IKG und Dr.-Ing. Franz Rottensteiner. Herrn Rottensteiner gebührt außerdem besonderer Dank für das ausführliche Korrigieren der ersten Entwürfe meiner Arbeit.

Ich möchte auch allen anderen Mitarbeitern am IPI danken für das hervorragende Arbeitsklima und die stets selbstverständliche gegenseitige Hilfe. Ich danke insbesondere Petra Helmholz für die schöne gemeinsame Zeit im Büro.

Dr.-Ing. Jan-Henrik Haunert (damals am IKG, heute am Lehrstuhl für Informatik I der Universität Würzburg) gilt mein Dank dafür, dass er mir geholfen hat, die Lineare Programmierung auf meine Problemstellung anzuwenden.

Der DFG möchte ich dafür danken, dass sie das Projekt gefördert hat, auf dessen Grundlage diese Arbeit entstanden ist. Außerdem möchte ich der Deutschen Gesellschaft für Photogrammetrie und Fernerkundung (DGPF) dafür danken, dass ich den Vaihingen-Bilddatensatz als Testdatensatz verwenden konnte.

Außerdem gilt mein Dank natürlich meiner ganzen Familie, besonders aber meiner Mutter und meinem Vater, die mir das Studium ermöglicht haben und mich immer auf vielfältige Weise unterstützt haben.

Curriculum Vitae - Lebenslauf

Persönliche Daten

Name	**Anne Grote**
Geburtsdatum	22.09.1979
Geburtsort	Berlin

Schulausbildung

August 1986-Juni 1999	Grundschule, Hannover/Espelkamp (1986-1990) Söderblom-Gymnasium, Espelkamp (1990-1999) Abschluss: Abitur

Auslandsaufenthalt

August 1999-Juni 2000	Fosen Folkehøgskole (Volkshochschule), Rissa, Norwegen (einjähriger Kurs in *naturliv* (Outdoor und traditionelles Handwerk))

Studium

Oktober 2000-Oktober 2005	Studium der Geodäsie an der Technischen Universität Berlin Abschluss: Diplom-Ingenieur Geodäsie Diplomarbeit in Zusammenarbeit mit der Bundesanstalt für Materialforschung und -prüfung (BAM) in Berlin („Geometrische Kalibrierung, Vergleich und Bewertung zweier Strahlablenk-Einheiten für die Verwendung in einem multispektralen Laserscanner“)

Beruf

seit Januar 2006	Wissenschaftliche Mitarbeiterin am Institut für Photogrammetrie und GeoInformation, Leibniz Universität Hannover